



*Editată cu minte, trup și suflet
pentru voi!*

REVISTA 2 ONLINE

NR. 14, martie 2021

INSPECTORATUL ȘCOLAR JUDEȚEAN GALAȚI

ASOCIAȚIA PROFESORILOR DE
INFORMATICĂ “TUDOR SORIN ” DIN GALAȚI

REVISTA DE INFORMATICĂ
(iM)PULS ++

NR. 14, martie 2021



Precizări terminologice

Cuvântul „reziliență” provine de la termenul latin „resilire”, care înseamnă „a ricoșa, a se întoarce”. Termenul de reziliență a fost preluat de experții din diferite domenii pentru a defini capacitatea unui sistem, a unei entități, a unei persoane de a se adapta rapid, prin antrenament, prin învățare, cu puțin stres, unui eveniment neplăcut, unei crize, unor probleme sau eșecuri.

Despre oameni, se spune că sunt rezilienți dacă știu să transforme un eșec într-o oportunitate și au capacitatea de a face acest lucru fără a acționa într-o manieră distructivă sau disfuncțională; dacă au capacitatea de a-și păstra sănătatea și energia când se află sub presiune constantă; dacă au capacitatea de a-și reveni rapid după eșecuri. În context organizațional, reziliența este măsurată ca fiind abilitatea de a face față oricărei situații care afectează buna funcționare. În majoritatea domeniilor de activitate, reziliența poate fi privită ca abilitatea de a reveni în forță după anumite schimbări sau evenimente stresante.

Context

Îmbunătățirea rezilienței sistemului de educație, prin planificarea unei reveniri la o educație de calitate pentru toți copiii, ar trebui să fie o prioritate maximă pentru următoarele luni și chiar ani și ar trebui să fie un principiu de bază al construirii unui învățământ flexibil, apt să facă față diferitelor provocări. De aceea, a vorbi despre modul în care pandemia a afectat educația și a accentuat inechitățile sociale existente, a evidențiat modul cum au fost afectați toți copiii, de la preșcolari la liceeni, înseamnă a veni în întâmpinarea a ceea ce îi așteaptă pe profesori, directori de școli, factori de decizie de la nivel local pentru a face față acestei provocări, al cărei impact s-ar putea resimți pe perioade greu de estimat.

Aspecte privind impactul psihologic al școlii online

În acest context, a aborda câteva [aspecte privind impactul psihologic al școlii online](#) la concurență cu procesul de revenire la cursul normal poate fi un demers util nouă, profesorilor, aceia care vom gestiona în fapt varietatea situațiilor apărute. Vom face referiri, în special, la segmentul de vârstă preadolescenței/adolescenței.

Despre efecte induse de „design”-ul lecțiilor online și al celor din sălile de clasă

Categoric, lecțiile „online” au avut/au nevoie de un design special și de o abordare pedagogică diferită față de cea din sălile de clasă, cu participare directă, ceea ce a determinat un antrenament diferit al proceselor psihice ale elevilor. Specialiștii susțin că în „online” creierul obosește mult mai repede. Însă, aspectul social al învățării nu a dispărut complet în cadrul cursurilor online, acesta s-a schimbat, doar: în loc să aibă discuții la clasă cu voce tare într-o sală de clasă fizică din școală, elevii se angajau în discuții virtuale în cadrul platformei de învățare online a cursului lor. La revenire, elevii vor avea nevoie să se readapteze la formatul clasic de predare. Cu toate acestea, se vor raporta la tipul de interacțiune virtuală, din care vor insera secvențe în conduita obișnuită, problemă la care profesorii va trebui să găsească soluții.

Despre consilierea părinților

Ce le putem recomanda părinților pentru perioada de readaptare ce urmează, în care adolescenții, mulți aflați în pragul unor examene importante din viața lor, se vor confrunta cu trecerea la normalitate? Cum pot însoți ei, părinții, proprii copii în toată perioada asta?



Deși adolescentul are capacitatea de a se organiza, de a-și crea un program, de a gestiona o situație nemaîntâlnită, de a se adapta la schimbări, totuși recomandarea este ca, în această perioadă, părinții să nu renunțe a-i fi alături, a-i facilita un program eficient care să implice studiul, sportul, momentele de relaxare și, constant, să-i reamintească adolescentului obiectivele majore, motivele pentru care și-a luat angajamentul de a învăța. De asemenea, este necesar să-i aprecieze efortul depus până acum și să-l orienteze spre ceea ce se va întâmpla după această perioadă, despre beneficiile ulterioare.

Despre viitorul școlii online

Poate deveni școala online o variantă a școlii viitorului? Cum se vede această schimbare din punct de vedere psihologic? Experiența prin care am trecut confirmă că școala online se poate face, excluzând pierderile sistemice, doar cu anumite grupe școlare (exclus grădinițe, primele clase de învățământ primar) și numai după o pregătire temeinică a profesorilor și a sistemului de învățământ pentru o asemenea abordare. Factorii de mediu familial pot fi perturbatori, influențând disciplina elevului, calitatea actului de învățare, starea emoțională și psihică. Predarea online se bazează pe comunicare și este necesar să evidențiem faptul că nu toți elevii pot înțelege informația la capacitate maximă. De asemenea, trebuie să luăm în calcul faptul că anumite materii, de genul orelor de sport, domeniul tehnic și profesional, activitățile muzicale, arta, dans și altele implică interacțiuni fizice. Predarea este un act empatic, oricât de performantă ar fi tehnologia, aceasta nu poate accesa o latură care stă la baza dezvoltării sănătoase a omului – acea latură emoțională în care nu pot fi transmise sentimentele, trăirile sufletești; nu se pot crea convingeri, așa cum se întâmplă într-un grup, unde stările se percep atât la nivel informațional cognitiv, dar și emoțional, iar acest lucru ajunge să fie o barieră în dezvoltarea lor în timp.

Concluzii

În timp ce pandemia a perturbat semnificativ predarea și a adâncit inegalitățile, aceasta a oferit, de asemenea, și oportunități pentru transformarea pedagogiei și transformarea etosului școlar: noi modalități de predare și învățare, noi moduri de comunicare cu copiii și familiile, noi roluri pentru menținerea ritmului general de pregătire a elevilor.

Școlile au multe de preluat din răspunsurile educaționale inovative la pandemie la nivel de școală, la nivel local și național. Au fost experiențe valoroase, vor fi învățate lecții importante, care se pot încorpora în procesele de planificare și reformare a educației pentru anii următori. Este o oportunitate care nu trebuie ratată, care necesită un proces de planificare menit să evalueze continuu situația, care intenționează să diminueze inegalitățile, care încurajează înregistrarea și documentarea proceselor de intervenție și a rezultatelor și care monitorizează și evaluează practicile, pentru a le îmbunătăți eficacitatea și a susține acțiunile reușite.

Repere bibliografice:

Biroul Regional UNICEF pentru Europa și Asia Centrală 2020 - *Crearea unor sisteme de educație reziliente în contextul pandemiei de COVID-19: Considerente pentru factorii de decizie de la nivel național, local și de unitate școlară.*

<https://www.g4media.ro/ce-ar-putea-fi-invatarea-online-in-lipsa-scolii-ruxandra-manea-antreprenor-in-educatie.html> (ultima accesare: 04.04.2021).



Digitalizarea școlii impune adoptarea strategiilor didactice specifice mediului online și adaptarea profesorului la schimbările desfășurării actului instructiv-educativ reflectată printr-o abordare specifică activităților educative ce necesită competențe IT. Abilitatea profesorului de a-și asuma implementarea actului didactic în online derivă din două trăsături de bază: deschiderea față de noutate și valorificarea conceptelor noi de tip informațional.

În anul 2000, doi profesori de chimie, Jonathan Bergmann și Aaron Sams, prin planificarea lecțiilor în online pentru elevii care nu puteau participa fizic la școală au observat că implicarea profesorului în predare este prezentă în special după parcurgerea conținutului științific de către elevi, aceștia putând studia independent materialul asincron. Cei doi profesori au pus bazele învățării mixte (atât offline, cât și online) - o formă de blended learning, numită Flipped Classroom.

Flipped Classroom – modelul **Clasei răsturnate** – este o abordare pedagogică în care instruirea directă se mută din spațiul de învățare de grup în spațiul individual de învățare. Este un nou concept care se bazează pe lucru individual, curiozitate epistemică, și capacitatea de a asimila noi cunoștințe. Modelul facilitează dezvoltarea autonomiei și gândirii critice, promovează acceptarea celorlalți și descoperirea punctelor slabe și forte care îi fac unici pe elevi. Această metodă poate să rezolve diminuarea abandonului școlar și să faciliteze învățarea pentru toți elevii. Implementarea și aplicarea conceptului Flipped Classroom (Clasei Inversate) are în vedere următoarele aspecte: resursele trebuie să fie accesibile tuturor elevilor pentru a facilita învățarea. metodele și procedura de îndrumare în procesul învățării.

Caracteristicile modelului Flipped Classroom

↳ Este **Mediu flexibil**

Se crează spații de învățare pentru găzduirea lecțiilor, utilizează aplicații, tehnologii IT. Se stabilesc spații și intervale de timp care permit elevilor să interacționeze și să reflecteze nevoilor de învățare, sprijină munca în grup, fie individuală și oferă modalități diferite de învățare a conținutului

↳ Dezvoltă **Cultura învățării**

Abordarea demersului didactic este centrată pe elev: elevul se angajează în activități de învățare fără prezența profesorului, devine autodidact, iar în prezența profesorului explorează subiectul în profunzime și creează

oportunități de învățare

Conținut intenționat

Profesorul determină ce materiale ar trebui elevii să exploreze independent și care este conținutul care va completa studiul individual al elevului. De asemenea creează un conținut accesibil și relevant pentru elevi

Educator profesionist

Profesorul antrenează elevii, le oferă feedback, are un rol mai puțin vizibil într-o "clasă răsturnată": urmărește progrese, corectează erori, exemplifică

Metodele modelului Flipped Classroom

↳ **Clasa standard inversată** – elevilor li se atribuie teme: de a viziona videoclip-uri, de a citi materiale relevante pentru ora următoare. În timpul orei, elevii practică ce au învățat prin activitățile școlare tradiționale

↳ **Orientat spre discuții** – profesorii atribuie videoclipuri sau materiale relevante pentru subiectul orei de curs, urmat de discuții și explorarea subiectului.

↳ **Orientat spre demonstrație** – se impune pentru situațiile în care elevului i se solicită să-și reamintească sau să repete activități, în aceste situații util este derularea unui videoclip și revizuirea activităților (memorie vizuală)

↳ **Metoda grupului** – există materiale/ resurse în spațiul virtual distribuite elevilor înaintea cursului. Elevii când revin la școală aprofundează noțiunile lucrând în echipă, această abordare îi încurajează pe elevi să învețe unii de la alții

↳ **Înlocuirea profesorului** – videoclipurile pot fi create și de elevi, astfel își demonstrează competența de a prezenta un subiect

Recomandări metodice:

↳ **Descoperă mediul virtual** – se recomandă să se cunoască tipul device-urilor folosite de elevi, viteza conexiunii la internet

↳ **Ține-l din scurt** – se recomandă ca subiectul să fie prezentat utilizând mai multe videoclipuri mici, se evită plictiseala

↳ **Urmărește participarea** – se recomandă ca profesorul să țină evidența participării elevilor astfel se asigură obținerea rezultatelor prin modelul Flipped Classroom

↳ **Întrebările elevilor** – se recomandă anticiparea întrebărilor pe marginea materialului pus la dispoziție elevilor pentru studiu și introducerea în material a răspunsurilor potențialelor întrebări.



Avantajele modelului Flipped Classroom

- ☞ **Mai mult timp unu la unu între profesor și elev** – crește timpul pe care trebuie să-l petreacă cu fiecare elev, creează o platformă unde elevii pun întrebări, are loc economisirea timpului din clasă din contul prelegerii în favoarea aplicării;
- ☞ **Mai mult timp de colaborare între elevi** – permite elevilor de a colabora între ei, o modalitate de a învăța în lucru în echipă
- ☞ **Elevii învață în ritmul lor** – în Flipped Classroom elevii învață în afara clasei, au mai multă autonomie în învățare
- ☞ **Elevii sunt încurajați să vină pregătiți la curs** – după ce elevii au studiat materialele digitale, revin în sala de clasă pregătiți cu idei și întrebări – o modalitate excelentă de a le spori simțul responsabilității
- ☞ **Conținutul poate deveni mai bogat** – elevii pot accesa mai multe surse sporind astfel înțelegerea subiectului, iar timpul alocat consolidării noilor noțiuni este mai mare, el fiind parcurs de elevi asincron.
- ☞ **Dezavantajele modelului Flipped Classroom**
- ☞ **Timpul crescut în fața monitoarelor** – device-urile conectate la internet tind să fie utilizate mai mult
- ☞ **Decalaj educațional** – pentru elevii care nu au acces la dispozitive electronice fără conexiune la internet
- ☞ **Timp și efort al profesorilor** – timpul și efortul necesar pentru crearea materialelor sunt mult mai mari decât în situația unei clase tradiționale
- ☞ **Mentalitatea neschimbată** – există profesori care nu se pot adapta și care vor încerca să transmită conținutul tradițional, neînțelegând importanța ”Clasei răsturnate”
- ☞ **Temă neefectuată** – pot exista elevi care nu acordă atenție materialului de studiat independent și revine la ora de curs cu temele neefectuate

Exemplu de Învățare hibrid cu modelul Flipped Classroom

Scenariu – profesorul este în clasă cu 15 elevi, ceilalți elevi sunt în online asincron conectați printr-o platformă educațională de web-conferencing și se aplică modelul Flipped Classroom. **Etapele de parcurs:**

Activitatea Profesorului - asincron

- își înregistrează explicațiile noilor concepte sau folosește un videoclip publicat (Youtube)
- se recomandă propria înregistrare deoarece

☛ profesorul adresându-se elevilor prin videoclip construiește și dezvoltă o relație strânsă cu elevii

☛ elevii sunt obișnuiți cu un stil de predare

☛ înregistrarea explicațiilor personale pot conduce la perfecționarea profesorului pentru că în momentul înregistrării este mult mai atent la ceea ce spune

☛ lecțiile deja înregistrate posibil să nu corespundă nevoilor/nivelului clasei de elevi

Activitatea elevului - asincron

elevii studiază materialul acasă, în ritm propriu.

(citește, vizionează, ascultă materialele postate)

elevii sunt încurajați să parcurgă în întregime materialul, să facă însemnări pe marginea lui, să scrie întrebări pe care le au și la care nu au găsit răspunsuri singuri, rezolvă sarcini date

elevi devin independenți în gândirea lor și își asumă mai multă responsabilitate în ceea ce privește învățarea.

Activitatea sincronă – la clasă

în clasă, în sistemul hibrid, are loc fixarea și consolidarea conținutului studiat acasă, asincron: în timpul orei (fizice, în clasă sau online, prin conferință video) discută despre ce au învățat sau citit.

elevii care nu și-au îndeplinit sarcina de studiu acasă, vor învăța de la colegii lor care au făcut-o deja.

profesorul răspunde la întrebările formulate de elevii din online sau de elevii din clasă, întrebări rezultate în urma parcurgerii asincrone a materialului

profesorul facilitează discuțiile cu elevii pornind de la întrebările lor și oferă informații suplimentare profesorul se poate concentra pe nevoile individuale ale elevilor, iar ei pot învăța în ritmul lor din discuțiile la clasă.

interacțiunea elev-conținut are loc în procesul inversat de învățare, când elevul parcurge noile informații independent.

interacțiunea elev-elev este asigurată prin grupele de lucru care pot fi formate din elevi din online și elevi din clasă, permițând astfel elevilor să funcționeze ca o clasă unitară

interacțiunea elev-profesor este asigurată în ora de curs, după parcurgerea conținutului, prin comunicarea dintre elevi și profesorii din clasă sau din mediul online

Concluzii - Modelul Flipped Classroom

Ideea de bază a tehnologiei „clasa inversată” constă în faptul că elevilor li se propune de a face cunoștință cu materialele lecției din timp, până la promovarea acesteia. Tema pentru acasă o reprezintă studiul independent al materialelor, care pot fi vizionate de atâtea ori, de cât este necesar elevului pentru a le înțelege și însuși. Pentru monitorizarea procesului materialele sunt însoțite de întrebări, în rezultatul răspunsurilor la care se determină nivelul reușitei. Reușita elevilor depinde, în mare măsură, de responsabilitatea celui ce studiază. Această formă de organizare a lecției oferă posibilitatea stimulării și orientării elevilor spre activitatea participativă reală.



P

erformanțele elevilor pot fi îmbunătățite dacă sunt învățați cum să-și organizeze informațiile. Organizarea informațiilor presupune analiză, sinteză, comparații, generalizare, abstractizare, concretizare, etc.

Procesul de predare-învățare algoritmică trebuie îmbinat cu învățarea euristică, elevii descoperind singuri algoritmi, iar după fixarea lor în memorie, aceștia urmând să fie aplicați în rezolvarea problemelor din aceeași clasă. În rezolvarea unei probleme, elevul trebuie să învețe să analizeze ipotezele problemei pentru a o încadra într-o clasă de probleme.

Acest proces activează cunoștințele anterioare după categoria de probleme. Încadrarea într-o anumită categorie de probleme este folosită în formularea ipotezelor privind modalitatea de rezolvare a problemei și, probabil, la estimări privind soluția problemei.

Pe măsură ce etape ale problemei sunt rezolvate, apar noi ipoteze în învățarea în continuare. Pe parcursul învățării elevul își poate revizui cunoștințele anterioare și își poate stabili obiective parțiale. Dificultatea în conexiunea dintre noile informații și informațiile anterioare poate apare dacă noile cunoștințe sunt neclare, dezorganizate sau lipsite de sens, dar și dacă subiectul este cointerestat în învățare.

Învățarea eficientă presupune:

- întoarcerea la informațiile anterioare pentru a verifica, compara, clarifica și a alege sau abandona strategii adecvate de învățare. Cel ce învață poate folosi aceste informații pentru a formula predicții, ipoteze despre înțelesul conținutului.
- integrarea noilor cunoștințe și conexiunea cu cele anterioare, generarea noilor cunoștințe. Nerealizarea unei conexiuni concrete duce la apariția lacunelor în cunoștințe și deci la lipsa unei strategii în dezvoltarea elevilor
- extinderea noilor cunoștințe și transferarea lor în alte situații.

Metoda algoritmizării constă în stabilirea unor pași în vederea rezolvării unor probleme și a asimilării pe această bază a cunoștințelor. A învăța să gândim înseamnă a înțel-

ege, a construi semnificații, a învăța să rezolvăm probleme, a construi modele aplicabile în diverse situații.

S-a arătat că în activitatea noastră zilnică, multe activități pot fi descrise într-o formă algoritmică, în sensul prezentării unor succesiuni de pași (etape) în realizarea lor. În realizarea oricărei probleme cu ajutorul calculatorului este necesară găsirea algoritmului în rezolvarea acelei probleme.

Deși noțiunea de algoritm a apărut încă din antichitate, ea este dezvoltată odată cu apariția limbajelor de programare.

Cuvântul algoritm provine din pronunția fonetică a numelui unui mare matematician arab pe nume Al-Khwarizmi Muhammed ibs Musa (780-850 d. Hr.).

Acesta a folosit pentru prima dată reguli precise și clare pentru a descrie procese de calcul (operații aritmetice fundamentale) în opera sa "Scurtă carte despre calcul algebric". Mai târziu, această descriere apare sub denumirea de algoritm în Elementele lui Euclid. Algoritmii lui Euclid pentru calculul c.m.m.d.c. a doua numere naturale este se pare primul algoritm cunoscut în matematică.

Inițial, noțiunea de algoritm a fost considerată primară. În matematica modernă noțiunea de algoritm a primit mai multe definiții: algoritmul normal al lui A.A. Markov, algoritmul operațional al lui A.A. Leapunov, mașina Turing/mașina cu stiva, funcții recursive, sisteme Post.

Prima descriere a definiției matematice pentru noțiunea de algoritm, a fost dată de matematicianul rus A. A. Markov.

În 1954, Markov definește noțiunea de algoritm ca "O regulă exactă care definește un proces de calcul ce conduce de la date inițiale variabile la un anumit rezultat". S-a demonstrat că din punct de vedere matematic aceste definiții sunt echivalente. În prezent, în domeniul probabilităților și statisticii, sunt cunoscute așa-numitele procese Markov. Astăzi se întâlnește foarte frecvent conceptul de proces sub diverse ipostaze: în reprezentarea și execuția algoritmilor, în funcționarea sistemelor de calcul, în execuția sistemelor de operare, în execuția programelor, în funcționarea rețelelor de calculatoare, etc.



Metoda algoritmică este foarte eficientă în procesul de învățare deoarece oferă elevului o anumită schemă de rezolvare a problemelor dar în același timp îl pune în situația de a fi solicitat în gândirea acestei scheme care să-l canalizeze în aflarea rezultatului. Există o strânsă legătură între demonstrație și algoritm. Aceste noțiuni se determină una pe cealaltă.

Calculatorul este o mașină care prelucrează automat informația. Pentru acesta trebuie să i se furnizeze datele pe care trebuie să le prelucreze ca fiind date de intrare și programul care ne spune cum să prelucrăm aceste date. Programul constă într-o succesiune de operații executate într-o anumită ordine, în urma cărora se vor furniza rezultatele ca date de ieșire. Rezolvarea unei probleme constă într-o succesiune de pași ce se desfășoară după un anumit algoritm, finalizându-se cu formarea de deprinderi ce pot fi extinse în alte situații mai complexe.

Și în rezolvarea unei probleme avem ipoteza sau parametrii inițiali, ca pornind de la aceștia și aplicând o succesiune de teoreme, axiome, legi etc., să se ajungă la o concluzie sau soluțiile problemei. În cadrul orelor de informatică, această metodă presupune folosirea materialului sub forma unui program elaborat pe principiul verificării și consolidării imediate a cunoștințelor, ceea ce permite corectarea operativă a erorilor comise și o autoreglare a învățării.

În elaborarea programelor se parcurg următoarele etape:

- precizarea obiectivelor operaționale;
- structurarea logică a conținutului după principiul pașilor mici și al învățării gradate;
- fracționarea conținutului în unități didactice;
- fixarea, după fiecare secvență, a întrebărilor, exercițiilor sau problemelor ce pot fi rezolvate pe baza secvenței informaționale însușite.

În găsirea unui algoritm de rezolvare a problemelor trebuie parcurse următoarele etape:

- se analizează datele de intrare și ceea ce cere problema, citindu-se de mai multe ori până când aceasta este clară;
- se analizează condițiile și se formulează, eventual, estimări privind rezultatul;
- se descompune problema în pași mici, se stabilesc succesiunile logice ale pașilor și apoi descrierea acestora;

-se testează algoritmul pentru mai multe date de intrare și se verifică dacă acesta conduce de fiecare dată la un rezultat.

Învățarea algoritmică nu este o metodă de sine stătătoare ci este o latură a fiecăreia din metodele cunoscute, fiecare dintre acestea având o etapă algoritmică. Necesitatea introducerii acestei metode în învățare apare datorită faptului că ea favorizează participarea efectivă și susținută a elevului la dezvoltarea proprie și deschide posibilități noi pentru activitatea de învățare.

Bibliografie

<https://gazeta.info.ro/conceptul-de-algoritm-abordare-moderna-i/>

Otilia Păcurari, *Strategii didactice inovative*, Editura Sigma, București, 2010.



Nu pot nega existența stresului dar trebuie să spun că modul în care te afectează, depinde în primul rând de arta de a contrabalansa principalele lui componente: cea subiectivă și cea obiectivă. Dacă faci parte din categoria celor care caută pete și-n soare, nu vei fi niciodată mulțumit de nimic. Iar dacă te-ai obișnuit să cauți vinovații pentru eșecul sau pur și simplu nemulțumirile și neîmplinirile tale, atunci vei fi nemulțumit de partenerii tăi: colegi, elevi, superiori, sistem etc.

Într-un microcosmos în care poți fi demiurg, dacă știi să te bucuri de orice pas înainte, dacă faci totul ca să miști lucrurile în sensul pozitiv al axei, e imposibil să nu reușești să dizolvi stresul în zâmbetul unui elev, în mulțumirea unui părinte. Depinde ce alegi ...

Și totuși de unde vine stresul ?

↳ din dezinteresul elevului pentru propria lui devenire

↳ din absența părintelui dintr-o ecuație în care el, ca parametru variabil influențează în mod determinant rezultatul

↳ dintr-o societate amnezică, guvernată de ceea ce am ales sau n-am ales

↳ dintr-o mass-media care promovează non-valorile și extrem de rar valorile, munca și efortul

În ceea ce privește efectul pandemiei asupra elevilor, pot spune fără teama că m-aș înșela că această perioadă ciudată și apăsătoare i-a determinat să găsească modalitatea perfectă de a se eschiva de la participarea activă la ore: profesorii nu au avut dreptul să le impună utilizarea camerelor video ... și de aici un fel de Tom și Jerry! Acolo unde profesorului i-a păsat de ceea ce face, de ce se întâmplă, după timide încercări, elevii năzdrăvani au capitulat pentru că au înțeles că nu merge! În special când profesorul era bine cunoscut pentru exigența sa, ca să nu spun așa cum ar spune ei: "al dracului". Au fost și situații fericite în care elevii au înțeles perfect care le este rolul, au colaborat și s-a putut lucra la fel de bine ca la clasă. Ba chiar aș îndrăzni să cred că pentru toți cei care au jucat corect și și-au dorit să facă școală adevărată, a fost ca și cum ar fi stat permanent în prima bancă. Parcă și relațiile profesor-elev s-au mai dezghețat, au căpătat alte valențe. Plutea în mediul virtual un fel de solidaritate, creată cu scopul de a înlătura efectele pandemiei.

Acolo unde este dragoste de carte, bun simț și interes de ambele părți se pot obține rezultate spectaculoase.

Nu tehnica, nu elemente de noutate împovărează, ci omul care nu e destul de OM!

Sunt profesor de informatică. Predau TIC, AutoCAD, Prelucrarea digitală a secvențelor audio-video (PDSAV) etc. În concluzie, la orele mele nu este suficient ca elevul să participe conectându-se de pe telefon, ci efectiv ar trebui să lucreze pe calculator. Cel puțin așa ar fi ideal! Nu pot însă să-i oblig să-și cumpere calculatoare ... și ei știu acest lucru. Ca să scape "mai ieftin", unii au declarat că nu au calculator și se conectează la lecții cu telefonul. Parțial adevărat! Până mai ieri, telefoanele performante – cele cu care navigă pe Internet - și mai ales abonamentul de date mobile erau prohibite prin prețul piperat; deci pentru jocuri, rețele de socializare și alte aplicații tocătoare de timp - foloseau PC-ul iar acum când ar fi nevoie de el, a dispărut fără urmă!

Știu – de la ei – agreează să intre la ore de pe telefon pentru că pot sta în pat, pot să se deplaseze în exteriorul locuinței ... și uneori chiar adorm (de ce nu?) pentru că nu e obligatorie camera. Doar prezența – că unii profesori nu se mulțumesc cu ce înregistrează aplicația și-i mai solicită să răspundă, interacționează destul de des în timpul unei ore. Deci: unde merge, merge ... unde nu, se conformează cerințelor!

Așa că la treabă! Cum să-mi conving elevii că școala în online e tot școală, că nu e în beneficiul nimănui să ne prefacem că facem școală?!

Cel mai ușor am rezolvat cu TIC-ul: există aplicații gratuite pentru telefon, le-au instalat și lucrează cu acestea. E mai dificil decât pe un PC, dar nu imposibil. Pentru sarcinile care nu pot fi rezolvate pe telefon, ca să nu-i păgubesc de punctajul aferent lor, am ales să partajez ecranul cu ei și să le ofer control pe device-ul meu să poată demonstra aici că știu cum se face. Dacă conexiunea e slabă atunci mă rezum la a le cere să mă dirijeze cum să execut eu operațiile necesare rezolvării sarcinii. Și așa au început să apară mai multe calculatoare, să scadă numărul elevilor care se conectează de pe telefon ...

Cum am procedat cu elevii de la Tehnician Proiectant CAD?

Prima problemă, generală și reală era softul! Nu ași fi putut să le sugerez să-și instaleze un soft piratat și nici să cumpere licență pentru un soft atât de scump! Mi-au venit în ajutor producătorii, adică cei de la Autodesk, cu licențele lor gratuite pentru studenți/elevi și profesori. Le-am vorbit elevilor mei, cu încântare despre generozitatea Autodesk-ului, despre posibilitatea de a ne instala pe calculatoarele noastre, ultima versiune de AutoCAD(...)



absolut gratuit pentru o perioadă de un an, cu posibilitate de prelungire ... Au început întrebările: care e adresa, cum procedăm, ce resurse tehnice sunt necesare? A, deci suntem pe drumul cel bun ... Mai rămâne problema calculatoarelor - nu avem calculatoare, ne conectăm de pe telefon! ... iar cei care nu au calculatoare - m-am gândit și la ei - vor face desenul pe hârtie, așa ca la "desen tehnic"! Asta pentru nota șase. Cine dorește mai mult, scrie pe foaia cu desenul și comenzile pe care ar trebui să le utilizeze pentru realizarea desenului în AutoCAD - ceea ce înseamnă încă un punct. Pentru alte două puncte, scrieți comenzile detaliate (cu parametri). Iar restul de un punct, este punctul pe care-l acord pentru aspectul estetic al lucrării și pentru perseverență!

Așa că proiectanții mei CAD, au început să-și "redescopere" vechile calculatoare, să le repare ... în final, să dovedească că au înțeles că e mult mai ușor și productiv să faci desenul unei piese pe calculator și nicidecum pe hârtie. Iată evoluția! Evident că nu am reușit cu toți dar sunt optimistă și perseverentă. Găsesec soluții și pentru ceilalți! Deocamdată, în timpul orelor de CAD, un elev (sau eu) desenează iar cei care nu au calculator, sunt "comentatori de pe tușă" - adică transmit celui care desenează comenzile, instrumentele pe care trebuie să le acceseze, calculează distanțele ...

Pentru cei de la multimedia, am avut o altă abordare. M-am plâns că elevii de la clasa la care sunt dirigintă, sunt cam delăsători - motiv pentru care aș vrea să fac pentru ora de dirigenție un scurt videoclip cu tema "procrastinarea" dar sunt într-o criză de timp și de ce să nu recunosc, poate și de idei?! Dacă ar putea ei, fiecare sau pe grupe de 2-3 elevi ... unii să se ocupe de sunet, alții de imagine, alții de montaj ...

Altă dată le-am cerut să-mi prezinte sub formă de videoclip cinci întrebări pe care le-ar pune la un interviu de angajare, unui potențial candidat.

Și toate repede, cu termen, cu note ...

Despre carte, despre lectură ... Da, desigur mi-am cumpărat câțiva zeci de metri liniari de carte, legată în piele eco, cu incrustații aurite ca să dea bine în bibliotecă! Evident - completează imaginea unui om citit, a unei case respectabile! ... și în felul acesta pot spune că am parte de aprecieri! Câți sunt atât de meticuloși și de subtili ca să mă verifice dacă într-adevăr am citit toți cei câțiva metri liniari de carte, pe care mi i-am înșirat frumos în lungul pereților? Dacă mai adaug că ei înșiși, ar fi trebuit să citească ca să mă poată interoga ...! Chiar aici aș putea completa cu cel puțin alte două celebre proverbe: "spune-mi cu cine te însoțești ca să-ți spun cine ești" și "râde hârb de oale sparte".

Să trecem la lucruri mai serioase! I-am întrebat pe elevii mei dacă mai cred în valabilitatea proverbului cu cartea ...

sau și acesta - ori e o utopie, ori un adevăr uzat moral?!

Ce mi-au răspuns?

De ce se bate atâta monedă pe studiu, pe carte?! Nu vedeți căți au 2-3 facultăți și stau tot pe banii părinților sau au plecat hai-hui în Europa și fac munci pentru care se cer alte abilități, cum ar fi: și începe un lung șir de "calități" ridicate la rangul de perfecțiune de către modelele actuale. Iar eu vin cu argumente ... susțin ideea de carte. "Lăsați doamna, că banii nu sar direct din carte!"

În același timp nu pot să nu mă gândesc că adevărul se zbate pe undeva, pe la mijlocul distanței dintre opiniile mele și ale lor, că sunt prea multe facultăți de tip "alba-neagra", că-i mai al dracului bac-ul decât facultatea! Și parcă doare versul lui Arghezi: "Carte frumoasă cinstea cui te-a scris!" . Aș vrea să-l parafrazez, să adaug și "cinstea cui te prețuiește"!

Concluzia: școala ar trebui reconsiderată, pusă într-o altă lumină ca să atragă, să ofere garanția succesului.

Cred eu, că nu pe toți dar pe o parte dintre ei, i-am convins că e mai ușor și categoric în avantajul tău - să fii cinstit, să cauți soluții constructive la probleme reale, nu la piedici inventate!

Întotdeauna am privit motivația învățării ca pe o ancoră puternică menită să-i confere stabilitate și în același timp flexibilitate celui supus modelării, creșterii prin învățare. Ca să-i poți adăuga plus valoare unui om, trebuie să-l de-termini mai întâi să-și dorească această transformare, să se deschidă și să fie capabil să o primească, conștient de ceea ce i se întâmplă. Să înțeleagă că așa cum ploaia udă florile dar creșterea este exclusiv treaba lor, așa și ceea ce-i oferă școala e doar o premisă a dezvoltării dar devenirea în sine, fomarea ca om și ca specialist într-un domeniu este strict rezultatul muncii, al efortului celui care învață.

Un om este motivabil atunci când ai reușit să-l înveți cât de importantă este stima de sine, de unde începe și unde nu se sfârșește vreodată!

Educație remedială în acest an școlar și în anul școlar viitor?

Profesorii care s-au implicat în susținerea orelor online au reușit să-i atragă pe acei elevi care sunt interesați de propria lor formare, adică aceiași elevi care la școală erau activi, implicați și obțineau rezultate frumoase. Lor li s-a adăugat și un eșantion redus de elevi timizi care în condițiile de desfășurare a orelor online s-au simțit în largul lor și astfel au înregistrat progres. Cei care fug însă de muncă în condiții reale, au făcut-o și în online - atât elevi cât și profesori din păcate. Și atunci de ce activități remediale? Evident, nu pot spune același lucru acolo unde sunt probleme tehnice legate de Internet, de aparatură ...

Deci ne-am jucat puțin de-a școala. Ce preț va avea aceasta, în economia anilor ce vin?! Cine poate ști, cine poate măsura ...



APLICAȚII PENTRU STIMULAREA PARTICIPĂRII DIVERSELOR TIPOLOGII DE ELEVII ÎN ONLINE

Prof. ANASIA Simona

Liceul Teoretic "Emil Racoviță" + Școala Gimnazială Nr. 7, Galați

Indiferent de contextul în care ne aflăm, fie el unul normal (cum este de dorit), fie unul mai puțin fericit (încerc să evit termenul de pandemic), un cadru didactic nu trebuie să renunțe la VISE. Și aici nu mă refer la dorințe, idealuri, scopuri, ci la un acronim simplu, frumos ordonat anagramatic în acest cuvânt, dar care este foarte greu de implementat în ritmul nostru zilnic de dascăli:

V = viziune (te gândești des unde vrei să vă aflați tu și tinerii cărora le predai în 3 – 5 ani);

I = influență (cu tot cu responsabilitatea și răbdarea asumării și însușirii feedback-ului);

S = statornicie (cu ținta urmărită, nu renunți când dai de greu);

E = energie (cu 30% mai multă oferită față de cea pe care o ceri de la elevii).

Și da, e posibil să nu ne reușească imprimarea acestui frumos termen în ADN-ul didactic, dar tocmai de aceea mai există un alt termen pe care am ajuns să îl apreciez din ce în ce mai mult. <https://youtu.be/XLeUvZvuvAs>

Scenariul hibrid ne provoacă pe toți cei implicați în procesul educațional, așa că personal consider că acum este momentul oportun pentru a ne apleca mai mult atenția către dezvoltarea emoțională a fiecăruia dintre noi și pentru a lucra în vederea stabilirii unui echilibru interior. Așa cum nu există profesori identici, elevii sunt și ei diferiți. Psihologii Thomas au formulat patru tipologii ale comportamentelor persoanelor (suplimentar față de CSFM – n-ai CSF J), pe care vi le supun atenției în următoarea prezentare. (DISC – dominant, influent, stabil, conformist / conștiincios).

Am început anul școlar cu un scenariu hibrid (o combinație între cel față în față și cel online), în care a fost (destul de) dificil să ne raportăm la tipologiile elevilor pentru predare – învățare, din cauza faptului că resursele (tehnologice și administrative) ale celor două grupe din clasă erau distincte. Acum, la trecerea 100% în online trebuie să ne gândim la modul în care elevilor le-ar face plăcere să lucreze de acasă. De aceea, nu putem pur și simplu să "transbordăm" lecția din sală în "meet-urile" zilnice la clasă, ci trebuie să adaptăm mijloacele de învățământ utilizate, ținând cont de faptul că după 5-7 minute de expunere elevii se plictisesc și este necesară interacțiunea și implicarea lor prin activități interactive. Astfel, intervin ruperile de ritm, extreme de folositoare în reenergizarea elevilor.

Exemple de ruperi de ritm:

<https://edpuzzle.com/media/5fb3cdcc100efc40c96ced63>

<https://ed.ted.com/search?q=computer>

[https://ed.ted.com/lessons/your-brain-can-solve-](https://ed.ted.com/lessons/your-brain-can-solve-algorithms-david-j-malan)

<algorithms-david-j-malan>

<https://wordwall.net/create/picktemplate>

<https://learningapps.org/12332203>

<https://learningapps.org/6115649>

<https://learningapps.org/createApp.php>

<http://mystery.knightlab.com/>

Am precizat mai jos o serie de linkuri la care puteți regăsi medii pe programare, exemple de material educațional în limba engleză, exemple de platforme pe care puteți crea lecții interactive.

https://www.onlinegdb.com/online_c++_compiler

[https://www.codechef.com/ide?](https://www.codechef.com/ide?itm_medium=navmenu&itm_campaign=ide)

itm_medium=navmenu&itm_campaign=ide

<https://www.khanacademy.org/computing>

<https://livresq.com/ro/>

<https://app.asq.ro/>

Alte aplicații

<https://www.onworks.net/> (Libre Office)

[https://sketchfab.com/search?](https://sketchfab.com/search?q=computer+components&sort_by=-relevance&type=models)

[q=computer+components&sort_by=-](q=computer+components&sort_by=-relevance&type=models)

<relevance&type=models>

Video

<https://youtu.be/Tgj6AZZjvys>

Iar pentru că viața e prea scurtă ca să o luăm prea în serios, propun să facem din fiecare zi de școală o experiență de joacă, în urma căreia să ne simțim suficient de provocați pentru încă o porție.



MATRICI PĂTRATICE

articol propus de profesor **Popescu Mădălina**
Colegiul Național "Mihail Kogalniceanu" Galați, Romania

Definiție

Un tablou bidimensional este tablou pătratic sau matrice pătratică dacă numărul de linii este egal cu numărul de coloane.

Date necesare prelucrare informații cu ajutorul unei matrice pătratică cu n linii și n coloane:

```
int a[100][100], i, j, n;
```

Exemplu de matrice pătratică:

	0	1	2	3	4
0	34	-5	14	9	87
1	87	48	-5	7	8
2	21	31	0	1	2
3	-8	5	8	-1	2
4	39	4	3	0	47

Elemente speciale într-o matrice pătratică:

Diagonala principală

Diagonala secundară

Elemente delimitate de diagonala principală

Elemente delimitate de diagonala secundară

Zonele delimitate de diagonale

Acces la elementele Diagonalei principale

	0	1	2	3	4	
0	34	-5	14	9	87	
1	87	48	-5	7	8	
2	21	31	0	1	2	
3	-8	5	8	-1	2	
4	39	4	3	0	47	

```
for(int i = 0; i < n; i ++)  
    "prelucrează" <<a[i][i];
```



Diagonala secundară

	0	1	2	3	4
0	34	-5	14	9	87
1	87	48	-5	7	8
2	21	31	0	1	2
3	-8	5	8	-1	2
4	39	4	3	0	47

```
for(int i = 0; i < n; i ++)  
    "prelucrează" a[i][n - 1 -  
i];
```

Observatie:

Dacă n este impar, cele două diagonale au un element comun.

Dacă n este par, cele două diagonale nu au elemente comune.

Elemente delimitate de diagonala principală

Deasupra Diagonalei principale					
	0	1	2	3	4
0	34	-5	14	9	87
1	87	48	-5	7	8
2	21	31	0	1	2
3	-8	5	8	-1	2
4	39	4	3	0	47

Se caracterizează prin faptul că indicele de linie este strict mai mic decât indicele de coloană $i < j$:

```
for(int i=0; i <n-1; i++)  
    for(int j=i+1; j<n; j++)  
        "prelucrează" a[i][j];
```

Sub diagonala principală					
	0	1	2	3	4
0	34	-5	14	9	87
1	87	48	-5	7	8
2	21	31	0	1	2
3	-8	5	8	-1	2
4	39	4	3	0	47

Se caracterizează prin faptul că indicele de linie este strict mai mare decât indicele de coloană $i > j$

```
for(int i=1; i <n; i++)  
    for(int j=0; j<i; j++)  
        "prelucrează" a[i][j];
```



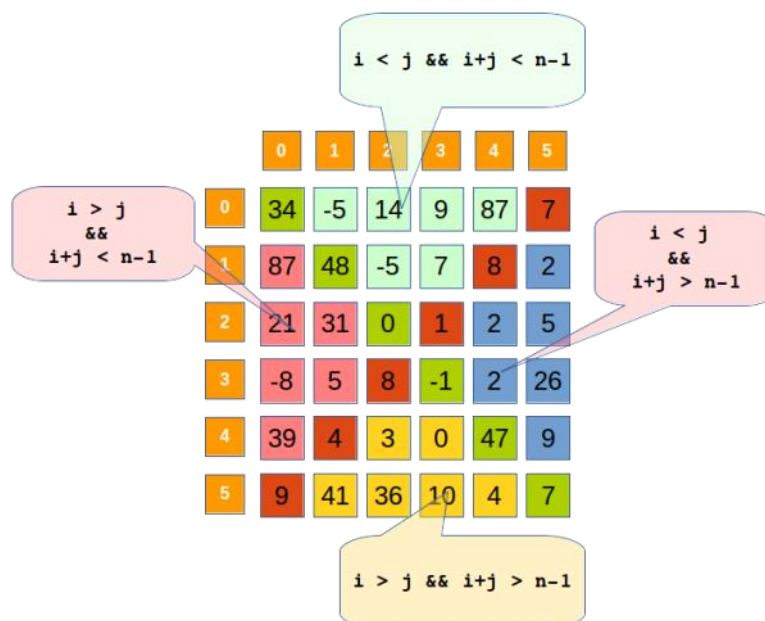
Elemente delimitate de diagonala secundară

<p>Deasupra Diagonalei secundare</p> <table border="1"> <tr><td></td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>0</td><td>34</td><td>-5</td><td>14</td><td>9</td><td>87</td></tr> <tr><td>1</td><td>87</td><td>48</td><td>-5</td><td>7</td><td>8</td></tr> <tr><td>2</td><td>21</td><td>31</td><td>0</td><td>1</td><td>2</td></tr> <tr><td>3</td><td>-8</td><td>5</td><td>8</td><td>-1</td><td>2</td></tr> <tr><td>4</td><td>39</td><td>4</td><td>3</td><td>0</td><td>47</td></tr> </table>		0	1	2	3	4	0	34	-5	14	9	87	1	87	48	-5	7	8	2	21	31	0	1	2	3	-8	5	8	-1	2	4	39	4	3	0	47	<p>Un element $a[i][j]$ situate deasupra diagonalei secundare se caracterizează prin faptul $i + j < n - 1$:</p> <p>//Vă invit să completați codul sursă</p>
	0	1	2	3	4																																
0	34	-5	14	9	87																																
1	87	48	-5	7	8																																
2	21	31	0	1	2																																
3	-8	5	8	-1	2																																
4	39	4	3	0	47																																
<p>Sub diagonala secundară</p> <table border="1"> <tr><td></td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>0</td><td>34</td><td>-5</td><td>14</td><td>9</td><td>87</td></tr> <tr><td>1</td><td>87</td><td>48</td><td>-5</td><td>7</td><td>8</td></tr> <tr><td>2</td><td>21</td><td>31</td><td>0</td><td>1</td><td>2</td></tr> <tr><td>3</td><td>-8</td><td>5</td><td>8</td><td>-1</td><td>2</td></tr> <tr><td>4</td><td>39</td><td>4</td><td>3</td><td>0</td><td>47</td></tr> </table>		0	1	2	3	4	0	34	-5	14	9	87	1	87	48	-5	7	8	2	21	31	0	1	2	3	-8	5	8	-1	2	4	39	4	3	0	47	<p>Un element $a[i][j]$ situate deasupra diagonalei secundare se caracterizează prin faptul $i + j < n + 1$:</p> <p>//Vă invit să completați codul sursă</p>
	0	1	2	3	4																																
0	34	-5	14	9	87																																
1	87	48	-5	7	8																																
2	21	31	0	1	2																																
3	-8	5	8	-1	2																																
4	39	4	3	0	47																																

Zonele delimitate de diagonala principală și secundară

Cele două diagonale delimitează în matrice patru zone: **Nord, Est, Sud și Vest.**

Condițiile verificate de indicii elementelor din aceste zone sunt prezentate mai jos:



Aplicație matrici pătratice:

Scrieți un program C++ care citește de la tastatură un număr natural n ($0 < n <= 25$) și construiește în memorie o matrice cu n linii și n coloane ale cărei elemente vor primi valori după cum urmează:

- * elementele situate pe diagonala principala vor primi valoarea 0;
- * elementele de pe prima coloana, cu excepția celui aflat pe diagonala principala vor primi valoarea n ;
- * elementele de pe a doua coloana, cu excepția celui aflat pe diagonala principala vor primi valoarea $n-1$;
- *
- * elementele de pe ultima coloana, cu excepția celui aflat pe diagonala principala vor primi valoarea 1.

Programul va afișa matricea astfel construită pe ecran, câte o linie a matricii pe câte o linie a ecranului.

timpul și visele...

*timpul și visele aleargă haotic
într-un carusel
de aer de apă
de vată de zahăr*

*1,2,3
și ne ascundem
într-un labirint
fără nicio ieșire
într-un calcul matematic fără soluție*

*între noi doar o tastă
transmite mesaje automate
despre naștere
renaștere*

*pielea mea filmată
cu raze x
nu acceptă coduri binare*

*de la șapte spre zece ani
tic-tac-urile sunt blocate în orologiile
vintage*

*milisecundele transmit impulsuri în
vene ultrasensibile
stimulând orice legătură dintre creier și
inimă*

*scrijelesc lumini pe inorogii
din caruselul cu numere mici și mari*

*soarele poate fi oglindit
cu o singură vocală*

Denisa Lepădatu
Clasa a XII-a, CNVA



Universul digital – informatica

Cu toții știm că programatorul este un creator de universuri pentru care doar el este responsabil. Universuri de complexitate nelimitată pot fi create sub forma programelor de calculator. Desigur că, programatorul este precum un copil care încearcă necunoscutul pentru a atinge absolutul.

Pentru a descoperi câteva taine ale informaticii avem nevoie de un îndrumător. Poate fi chiar telefonul din fața noastră, calculatorul, dar și un profesor care are o metodă originală de transmitere a informațiilor .

Dragi cititori, vă pot spune cu încredere că domeniul IT este precum o rețetă, principalele ingrediente fiind pasiunea și motivația, iar apoi presăram puțin praf de stele: ambiție și motivație.

Nu de mult timp, am avut ocazia de a conversa cu doi olimpici la informatică, Marius Iordache și Sorin Dincă, foști „Cuziști”, care ne-au împărtășit experiențele lor din cariera de IT-iști.

„Nu există o rețetă a succesului [...], trebuie doar să iei lucrurile în serios [...], important e să îți placă ceea ce faci”, ne-au declarat cei doi .

Am reținut faptul că aceștia au fost mereu motivați și implicați, dezvoltând noi abilități practice și cunoscând/descoperind noi ramuri ale acestui domeniu. Ca adolescenți, cu siguranță au avut și momente de declin, dar perseverența lor a fost mai puternică, mai accentuată decât gândul de a abandona pasiunea lor.

Interesantă a fost afirmația lor :

„Să riști când ai o oportunitate, pentru că după, vor veni și recompense”, evidențiind faptul că experiența este foarte importantă și căutată de angajatorii companiilor IT, idee care cred că ar trebui să devină ca o „deviză” pentru tinerii care se pregătesc în orice domeniu.

Cât despre concursul „Severin Bumbaru”, Marius și Sorin ne-au mărturisit că atmosfera din acele timpuri era relaxantă, și că ei participau cu drag, ca și cum realizau o activitate prietenească, la ei acasă. Acest concurs le-a dezvoltat creativitatea și imaginația, explicându-ne cât de important este exercițiul suplimentar pentru a atinge un nivel satisfăcător, atât la concursuri, cât și la olimpiade.

O întrebare adresată de noi, a fost dacă informatica poate fi învățată de oricine, răspunsul a fost unul concret și simplu: “Da, la fel cum oricine poate să citească și să scrie, la fel poate căpăta cunoștințe de informatică, cum ar fi să folosească un calculator, să acceseze internetul, să trimită un email, să scrie un text, etc.”

O ultimă observație sesizată de multe companii, este recrutarea tinerilor în această lume plurivalentă. De multe ori este o provocare, dar dacă aspiri să fii cel mai bun, iar o cariera de specialist IT îți surâde, trebuie să urmezi o linie continuă de pregătiri ca să te poți dezvolta și adapta în mod constant la cerințele de pe piața muncii și nu numai. Dar ca să putem vorbi despre procese tehnologice de la macro la micro și apoi la nano trebuie să privim asupra elementelor fundamentale: limbajele de programare!

În concluzie, consider că un om destinat acestei lumi trebuie să fie creativ, dispus să își asume riscuri, să aibă o gândire logică, să fie mereu deschis pentru a învăța lucruri noi, care, la rândul lor, îi vor deschide noi perspective către perfecționare. Totul implică efort și dăruire, ceea ce ușurează mult „urcușul” într-un domeniu care nu de mult timp „și-a făcut loc” în ceea ce se numește carieră.

Elena Moraru, clasa a X-a
Nicoleta Simona Ene, clasa a X-a





Structuri repetitive în Scratch

Prof. Adriana Ciocănel
Colegiul Național "Calistrat Hogaș"

Despre Scratch

La nivel internațional, pentru familiarizarea copiilor cu limbajele de programare, se folosește intensiv Scratch, limbaj de programare creat de MIT Media Lab. Sloganul lor, "Imagine. Program. Share", este materializat prin faptul că mediul de programare are o interfață prietenoasă, permițând o navigare ușoară și intuitivă deoarece blocurile sunt grupate pe categorii de acțiuni. În plus, atunci când copiii realizează proiecte în Scratch, ei învață elemente importante de programare; în același timp se pot manifesta creativ și lucrează colaborativ.

Introdus acum patru ani în programa școlară în cadrul orelor de Informatică și TIC, pentru clasele a Va și a VIa, Scratch se dorește a face transferul către programarea clasică, punând bazele unei gândiri logice și structurate, fără ca elevul să se lovească de erori și sintaxă. Pentru a da însă roade, gândirea elevului trebuie însă orientată cu atenție de către profesor.

Structuri repetitive

Atunci când o operație sau un grup de operații trebuie repetate, este preferabil să se folosească un bloc care permite repetarea acestora, față de situația în care am scrie repetat, individual, aceste operații. În plus, numărul de repetiții ar putea să difere în funcție de nevoi, lucru pe care nu am putea să-l gestionăm prin scrierea repetată, individuală, a acestor operații.

În Scratch găsim blocurile care implementează structurile repetitive în grupul Control, iar structurile repetitive sunt următoarele:



Implementează structura repetitivă cu număr cunoscut de pași; în imagine, eventualele acțiuni sunt repetate de 10 ori.

Acest ciclu se repetă la infinit. Cum repetarea la infinit a unei acțiuni sau a unui grup de acțiuni nu permite atingerea rezultatului, în interiorul său vom avea o condiție sau o instrucțiune de ieșire ca, de exemplu, "oprește script" sau "oprește tot".

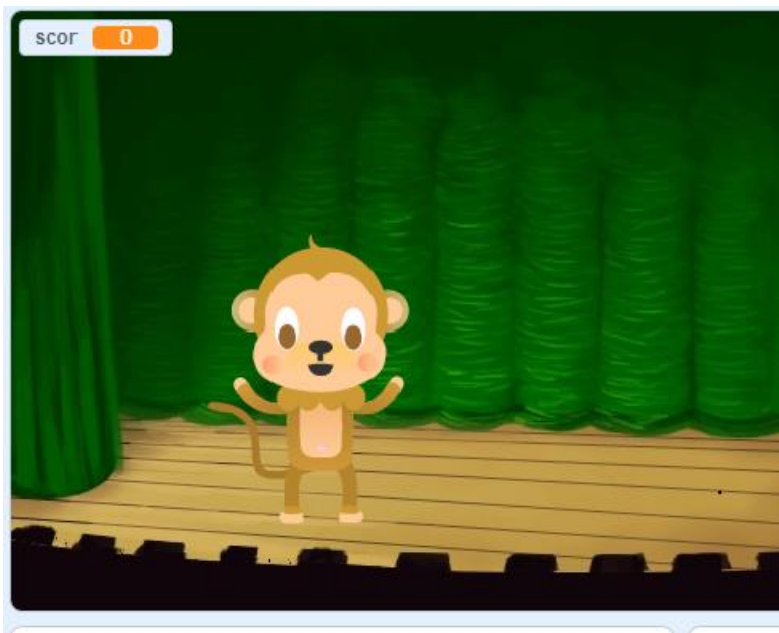




Acest ciclu se execută de 0 sau mai multe ori, până când condiția este adevărată. <> reprezintă o condiție care se evaluează la fiecare început de ciclu.

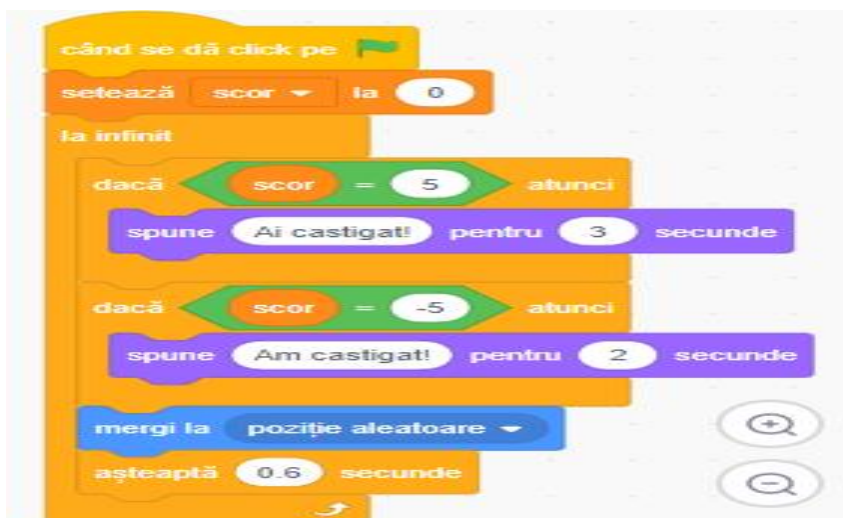
Aplicații propuse

Aplicația 1 Realizați o aplicație sub forma unui joc care să conțină un personaj animat. Când vom atinge personajul cu un click de mouse, va crește punctajul înregistrat în joc cu 1. Dacă vom efectua un click de mouse în scenă,



punctajul va fi decrementat cu 1. Se va seta un punctaj maxim și un punctaj minim ce poate fi atins de către jucător.

Script personaj



Analiza problemei: se observă din enunț că modul de finalizare a jocului este determinat de punctaj. Din acest motiv vom defini variabila **scor**, pe care o vom inițializa cu 0 pentru fiecare nou joc. Am ales maxim de scor egal cu 5, caz

Script scena

în care câștigă jucătorul și minim de scor

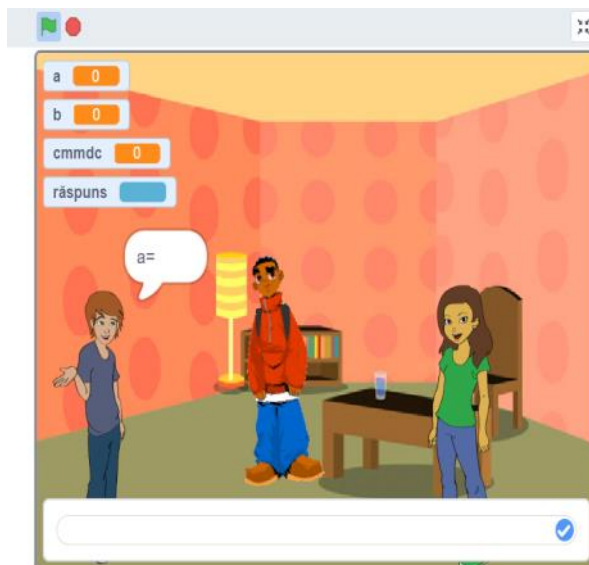
egal cu -5, caz în care câștigă personajul.

Link: <https://scratch.mit.edu/projects/492115999>



Aplicația 2 Implementați în Scratch algoritmul de calcul al cmmdc a două valori. Veți concepe aplicația ca un dialog între trei personaje, în care primele două personaje vor furniza cele două valori pentru care se determină cmmdc, iar al treilea personaj va furniza valoarea cmmdc.

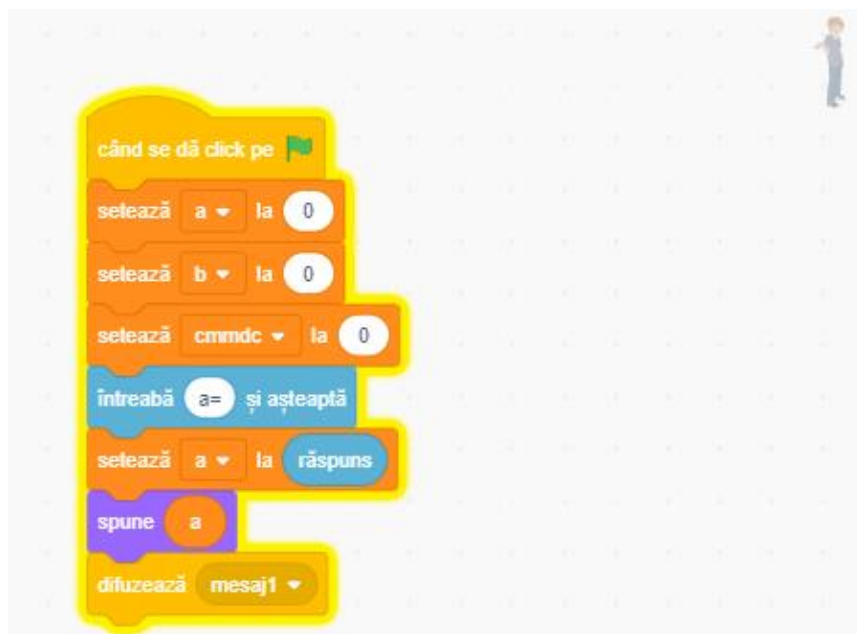
Observații Aplicația propusă va fi precedată de discutarea și prezentarea algoritmului de determinare a cmmdc a două valori prin scăderi repetate. Pentru ca elevii să înțeleagă



algoritmul, consider că este necesar ca după prezentarea acestuia să urmeze și un exemplu numeric cu desfacere pe pași pentru a înțelege condiția de oprire a structurii repetitive.

Analiza problemei – se vor identifica împreună cu elevii datele de intrare (două variabile a și b) și datele de ieșire (cmmdc).

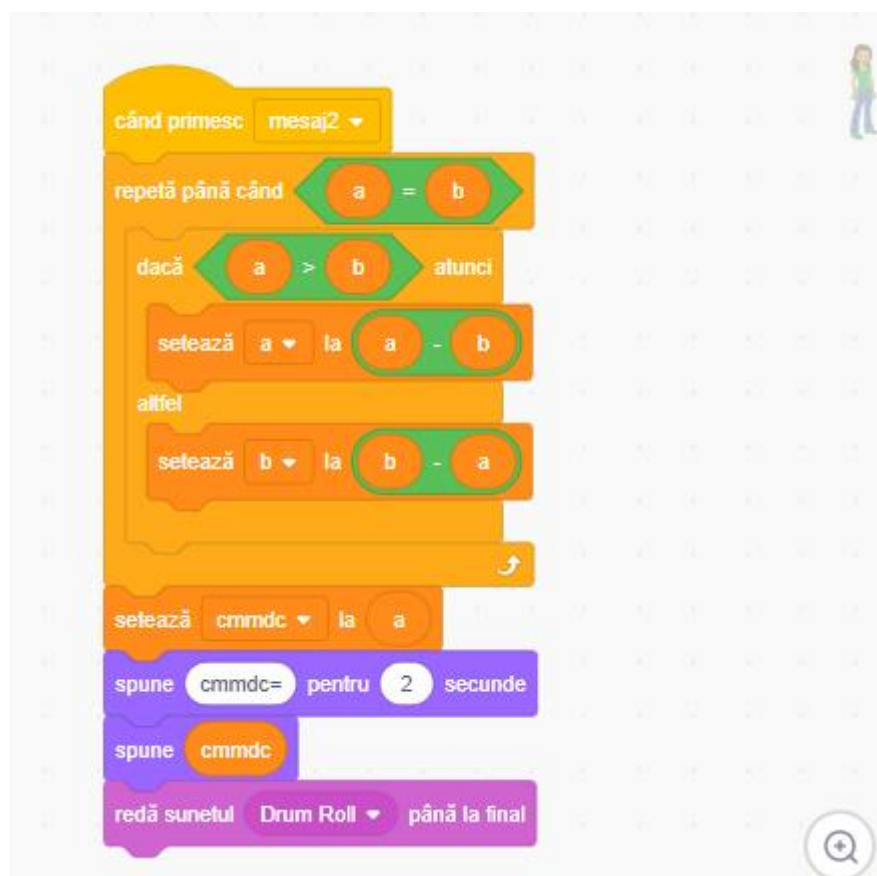
Script personaj 1



Script personaj 2



Script personaj 3 – implementează algoritmul de calcul al cmmdc



Link: <https://scratch.mit.edu/projects/499574337/>



La momentul actual, programarea calculatoarelor este unul dintre domeniile de activitate cu cea mai rapidă rată de dezvoltare. Dacă inițial vei căuta să înveți programare doar pentru că e cool sau pentru că ești în căutarea unui job de developer care să îți ofere stabilitate financiară, ulterior s-ar putea să faci din aceasta un stil de viață care îți va oferi mijloace noi de a comunica și de a te conecta cu alți oameni, să constăți că vei face parte dintr-o cultură într-o continuă evoluție, că te vei dezvolta și transforma.

Înainte de a alege primul limbaj de programare ar trebui să te întrebi ce calități îți sunt necesare ca să înveți informatică? Ei bine, ar trebui să știi că ai nevoie de logică, de motivație și de creativitate. Practica, determinarea și răbdarea îți vor fi cei mai buni aliați și din fericire în acest domeniu există o multitudine de resurse valoroase și da, google poate fi un ajutor de nădejde.

Revin spunând că vei avea nevoie de multă – multă răbdare deoarece învățarea nu va fi un proces liniar prin care la sfârșitul unei zile vei avea certitudinea că ai bifat cele 3 capitole propuse la începutul săptămânii. Uneori vei avea senzația că stagnezi sau că ești prins într-o buclă. Este important să nu te descurajezi și să ai încredere că vei depăși acel moment.

Adeseori cei care vor să învețe programare se întreabă **cu ce limbaj de programare să încep? Care este cel mai bun limbaj de programare?**

Trebuie să știi că cel puțin la început logica unui program nu depinde prea mult de limbajul de programare pe care îl vei învăța. Gândește-te că atunci când rezolvi o problemă la matematică nu are importanță dacă gândești soluția în limba ta maternă, în engleză sau în chineză. De aceea îți recomand să parcurgi inițial măcar la modul elementar câteva noțiuni care țin de **reprezentarea algoritmilor în pseudocod** și poți începe cu **structurile de control**. Astfel îți vei putea organiza clar ideile, vei ști ce ar trebui să faci ulterior programul tău, de câte ori și în ce context. Limbajele de programare sunt foarte riguroase în sensul că au niște reguli de scriere (sintaxă) care dacă nu sunt respectate pot genera erori. **Limbajul pseudocod** face o tranziție de la limbajul natural la limbajul de programare și e așa, un soi de limbaj universal care se poate transcrie / traduce în orice limbaj de programare. Poate părea anost la început să gândești o aplicație folosindu-te de foaie și pix dar crede-mă că cei mai buni programatori nu au nevoie de un limbaj de programare ca să-și gă-

ndească soluțiile. Limbajul de programare este doar un instrument care traduce ideea ta exprimată în pseudocod într-un limbaj pe care sistemul de calcul îl poate interpreta.

Cred că se poate deduce din cele de mai sus că nu există un limbaj de programare mai bun decât celelalte. De aceea după ce ai parcurs structurile de control în limbajul pseudocod ți-aș sugera să alegi un singur limbaj de programare prin care să experimentezi câteva etape:

Învăță sintaxa limbajului, chestiunile de fundamentale (cum preiau datele, cum modific date cum generez rezultate)

Transpune în limbajul de programare ales structurile de control învățate în pseudocod, mai exact studiază instrucțiunile limbajului

Încearcă să creezi niște programe simple care rezolvă probleme din viața cotidiană. De exemplu să determini media unor valori, valoarea unui coș de cumpărături, cel mai bun rezultat sportiv, un joc simplu de tip X și 0 etc. Vor trece câteva săptămâni bune de studiu până aici. Nu te așteptă ca după o săptămână să faci un joc.

Învăță să te joci cu datele, să le ordonezi, să le schimbi ordinea, să stochezi rezultatele. Uite, până aici îți vei exersa bine logica.

E momentul să faci un upgrade programelor tale jucându-te cu interfața aplicațiilor. Probabil ai auzit de termenul de user-friendly.

Dacă ai parcurs etapele de mai sus ai suficiente cunoștințe să îți exprimi creativitatea. Încearcă să dezvolti o aplicație mai avansată. Integrează în proiectul tău soluțiile celorlalți, secvențe de cod care îți rezolvă o parte din cerințe. Nu e necesar să reinventezi roata dar e important să înțelegi ceea ce fac ceilalți.

Studiază, aprofundează, rafinează. Ai ajuns în punctul în care poți să compari soluții, în care vei constata că primele tale soluții erau extrem de naive și că rezolvi totul mult mai ușor. Acum abia poți înțelege deplin puterea limbajului pe care l-ai ales și cât mai multe dintre facilitățile pe care ți le oferă.

Integrează și alte tehnologii. Lucrează cu cineva la un proiect nou.

Revenind la întrebarea *cu ce limbaj de programare să încep* chiar dacă nu există un limbaj universal valabil pentru orice începător îți poți însă analiza intențiile pe viitor, dacă te interesează partea de game development, mobile web development, systems programming etc.



Oricum, o dată ce ai reușit să parcurgi etapele de mai sus într-un limbaj de programare vei constata că vei parcurge aceleași etape mult mai ușor când vei trece la un alt limbaj pentru că **gândirea algoritmică este independentă de limbajul de programare.**

O listă cu cele mai populare limbaje de programare din 2021, căutate nu doar în întreaga lume, ci și în România te va orienta în alegerea primului limbaj de programare:

1. Limbajul de programare Python - este unul dintre cele mai ușor de învățat și utilizat limbaje de programare pentru că are puține particularități și o sintaxă simplă, care poate fi înțeleasă cu ușurință, chiar și de cei fără experiență în codare. Acesta poate fi folosit atât pentru a crea aplicații pentru desktop, cât și aplicații web sau framework-uri.

2. Limbajul de programare C - un limbaj de programare "old school", printre primele apărute, C e la fel de relevant astăzi ca în momentul în care s-a lansat, în 1970. C este un limbaj de programare procedural, flexibil și versatil, care îi permite programatorului să aibă control maxim prin comenzi minime. Limbajul C are foarte puține cuvinte cheie și structuri semantice, însă nu este foarte ușor de învățat, mai ales pentru un începător în programare - pregătește-te pentru o curbă de învățare mai mare, dar care va merita din plin toate eforturile tale.

Pentru că multe persoane optează pentru limbaje de programare mai noi, învățarea limbajului de programare C te poate diferenția pe piața de muncă pentru că C rămâne an de an în preferințele angajatorilor, mai ales dacă vrei să lucrezi în game development și systems programming.

3. Limbajul de programare C++ - este o versiune actualizată a limbajului C. C++ este un limbaj scalabil, cu ajutorul căruia poți crea aplicații ce consumă multe resurse, precum aplicații pentru desktop sau jocuri cu grafică complexă. Așa că dacă vrei să lucrezi ca developer în industria de gaming, C++ este limbajul potrivit pentru tine.

4. Limbajul de programare Java - este un limbaj de programare orientat pe obiecte, care a fost lansat în 1995 și care este astăzi cel mai popular limbaj de programare din lume, cât și cel mai stabil. Acesta poate fi utilizat pentru aplicații de desktop cu interfață grafică, aplicații de mobil (mobile web development) sau aplicații și chiar servere web, fiind foarte versatil și compatibil cu multe platforme.

5. Limbajul de programare PHP - este un limbaj de programare special creat pentru web development. PHP este deopotrivă un limbaj simplu de programare pentru începători, însă în același timp îi oferă unui de-

veloper experimentat multe funcționalități avansate.

Chiar dacă există developeri care consideră că PHP un limbaj de programare învechit, acesta își păstrează relevanța chiar și în 2021. Conform Best Programming Language, 80% dintre cele mai bune 10 milioane de site-uri din lume folosesc PHP.

6. Limbajul de programare JavaScript - este un limbaj de programare cu ajutorul căruia poți construi site-uri interactive. Este una dintre tehnologiile fundamentale ale internetului, alături de HTML și CSS.

7. Limbajul de programare C# (C sharp) - este un limbaj de programare orientat pe obiecte, ce a fost lansat în 2000 și gândit ca competitor pentru limbajul Java, cu care este foarte similar.

C# este un limbaj de programare versatil, cu care poți crea aplicații pentru Windows, dar și aplicații pentru web sau pentru device-uri mobile. C# are cea mai rapidă evoluție și este unul dintre limbajele folosite pentru a crea aplicații în framework-ul .NET.

8. Limbajul de programare Swift - Creat de Apple în 2014, Swift este un limbaj de programare compatibil cu sistemele de operare și framework-urile Apple. Limbajul Swift este similar cu Python și Ruby și a fost gândit să fie ușor de utilizat pentru începătorii care aspiră să devină software engineer.

Atunci când faci o alegere e necesar să ții cont de specificul limbajului, de cunoștințele tale în domeniu, dar și de diversitatea ofertelor de pe piața de muncă la care îți oferă acces. În momentul de față platformele de joburi din România pun accentul pe Java, C++ și PHP.



App Inventor sau despre cum am realizat prima aplicație mobilă

*Delia Maria Paraschiv și Mihail Stanciu, clasa a VII-a
Școala Gimnazială „Iorgu Iordan” Tecuci*

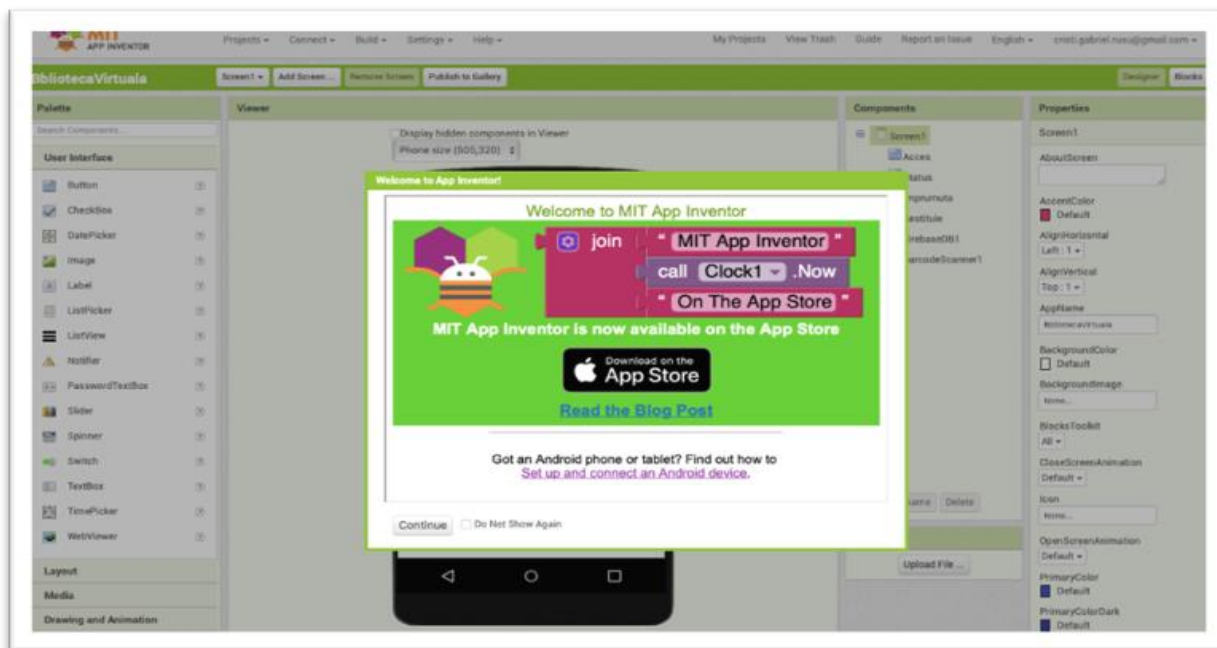
MIT App Inventor este un mediu de programare vizual intuitiv care permite tuturor - chiar și copiilor - să construiască aplicații complet funcționale pentru smartphone-uri și tablete. Fără o experiență prealabilă, folosind App Inventor se poate realiza o aplicație simplă și funcțională în mai puțin de 30 de minute. În contextul în care programarea se învață în mod vizual, acest instrument bazat pe blocuri facilitează crearea de aplicații complexe, cu impact ridicat, într-un timp semnificativ mai mic decât mediile de programare tradiționale. Proiectul MIT App Inventor încearcă să democratizeze dezvoltarea de software printr-un mediu grafic ofertant, destinat tuturor oamenilor, în special a tinerilor, făcând trecerea de la consumul de tehnologie la crearea de tehnologie.

Inițiativa acestei platforme, care are la momentul actual mai mult de 6 milioane de utilizatori înregistrați, a pornit de la o mică echipă de angajați și studenți CSAIL, condusă de profesorul Hal Abelson.

Mediile grafice de codare bazate pe blocuri stimulează capacitatea intelectuală și creativă. MIT App Inventor este mai mult decât atât și reușește să ofere copiilor abilități reale de a face diferența, constituind o modalitate de a obține un impact social vizibil de o valoare incomensurabilă pentru comunitate.

Cum am început noi să folosim această aplicație? Am pornit de la un concurs, Hardcore Hackathon care s-a desfășurat în mediu online, având o etapă de inițiere, una de mentorat din partea profesorului de informatică Cristi Rusu de la Școala Gimnazială „Iorgu Iordan” Tecuci și o fază competițională în care trebuia să prezentăm aplicația propriu-zisă, un plan de afaceri și o descriere sub forma unui Pitch, care să convingă investitorii să ne finanțeze. Timp de două săptămâni, în două week-enduri, am urmărit pe platforma Microsoft Teams 4 webinaruri despre dezvoltarea aplicațiilor mobile, folosind Android Studio și AppInventor, un webinar despre realizarea planului de afaceri și un altul despre public speaking.

În etapa de mentorat am pornit cu platforma AppInventor care este destul de ușor de utilizat și nu necesită resurse foarte mari, rulând online. Ea este disponibilă la adresa <http://appinventor.mit.edu> și necesită instalarea pe telefon a unui emulator (MIT AI2 Companion) fie instalarea pe calculator a unui simulator al telefonului. De menționat că atât telefonul cât și calculatorul de pe care accesăm platforma AppInventor, vor trebui să fie logate în aceeași rețea de internet. Pentru a realiza o aplicație, este nevoie pur și simplu să accesăm Create Apps! Și să ne logăm cu un cont de gmail sau să ne realizăm un cont pe această platformă. Odată autentificați pe platformă (vezi fig. 1) ni se propune aplicația pe care va trebui să o instalăm pentru a rula proiectul nostru.

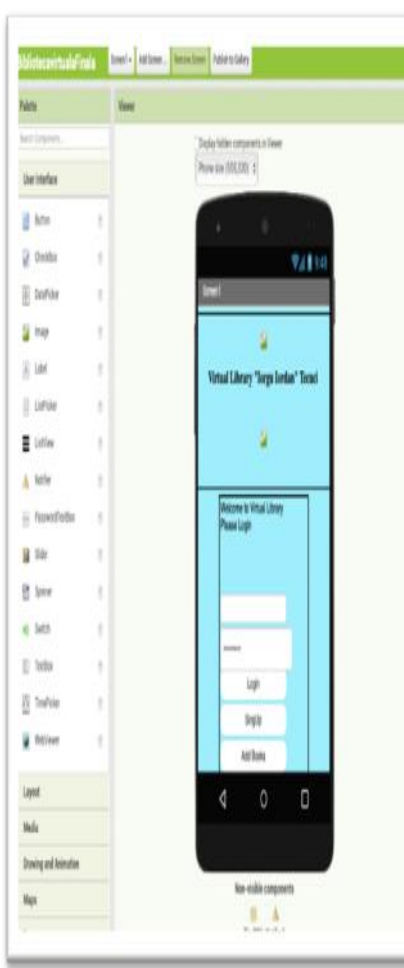


Facem mențiunea că aceste aplicații nu vor rula chiar dacă le vom transfera pe telefon, ci se vor executa prin această aplicație mobilă. Pentru a realiza aplicații mobile de sine stătătoare, va trebui să folosim fie Android Studio fie Dart-Flutter-Python/Flask, etc pentru a realiza produse mobile mai avansate și care să ruleze autonom pe dispozitive smartphone/tabletă.

Aplicația este disponibilă în 16 limbi străine și este în atenția noastră să realizăm o traducere a interfeței, în colaborare cu profesorul nostru de informatică. În cadrul concursului, care a fost și unul din motivele pentru care am redescoperit această platformă, cu care ne-am întâlnit cu trei ani în urmă în cadrul unor activități și Săptămâna Școala Altfel, am aflat cât de simplu este să realizezi butoane și meniuri și apoi cu ajutorul funcțiilor să faci să prindă viață aplicația mobilă.

Idea noastră a fost să digitalizăm accesul în bibliotecă, prin realizarea unei aplicații mobile care să permită autentificarea utilizatorilor în baza de date a bibliotecii, împrumutul cărților și restituirea acestora. Pentru aceasta am realizat mai multe ecrane, de autentificare, de administrare, de împrumut, de restituire și de consultare a bazei de date.

În cele ce urmează vom prezenta secțiuni de cod și modul de lucru. Am început prin *a seta un layout* de tip tabel în care am plasat prin drag-and-drop elemente vizuale (butoane, barcode scanner)



```
initialize global Username to ''
initialize global Password to ''

when ButtonLogin Click
do
  set global Username to call TinyDB1 GetValue
  valueTagOfThere tag TextBoxUsername Text
  set global Password to call TinyDB1 GetValue
  valueTagOfThere tag PasswordTextBox1 Text
  if
    get global Username == Username and get global Password == Password
  then
    open another screen screenName Screen1
  else
    call NotificationCenter ShowAlert
    notice Incorrect login click SingUp Button

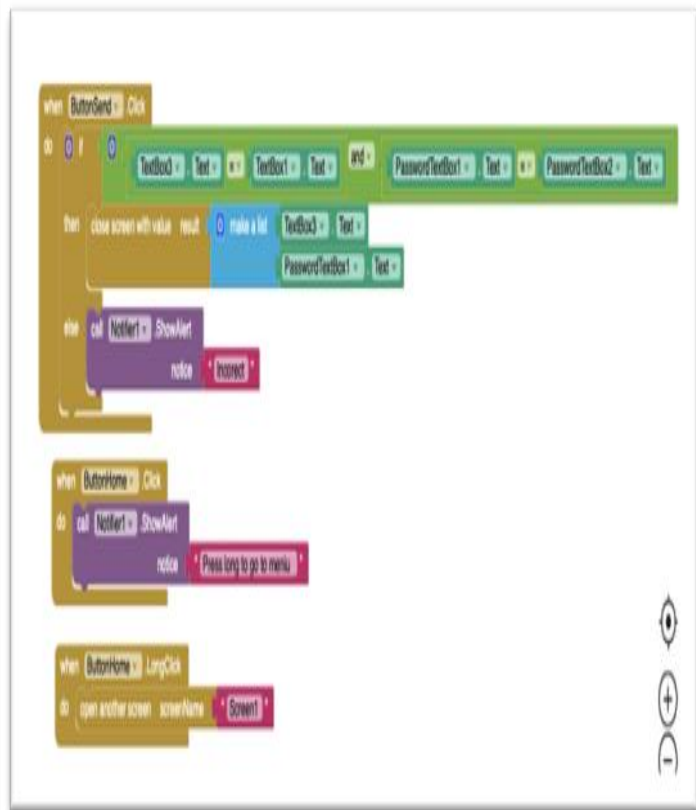
when ButtonSingUp Click
do
  open another screen screenName Screen2

when Screen1 OtherScreenClosed
otherScreenName result
do
  call TinyDB1 StoreValue
  tag select list item list get result
  valueToStore Username
  call TinyDB1 StoreValue
  tag select list item list get result
  valueToStore Password

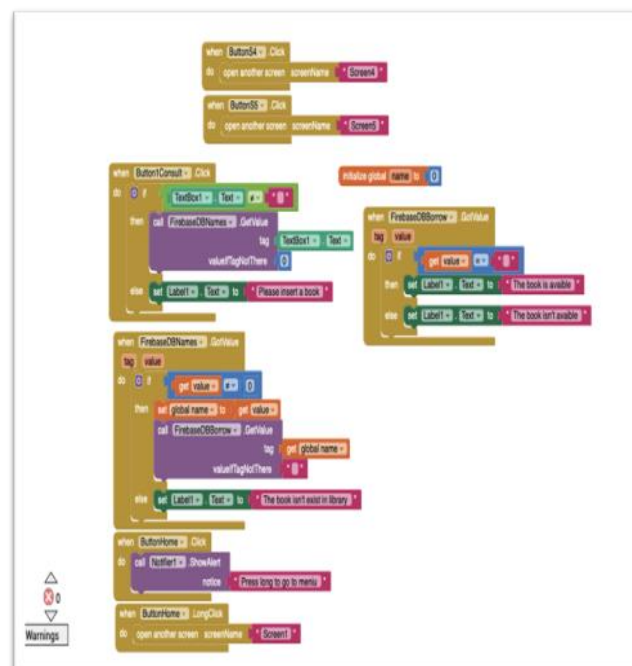
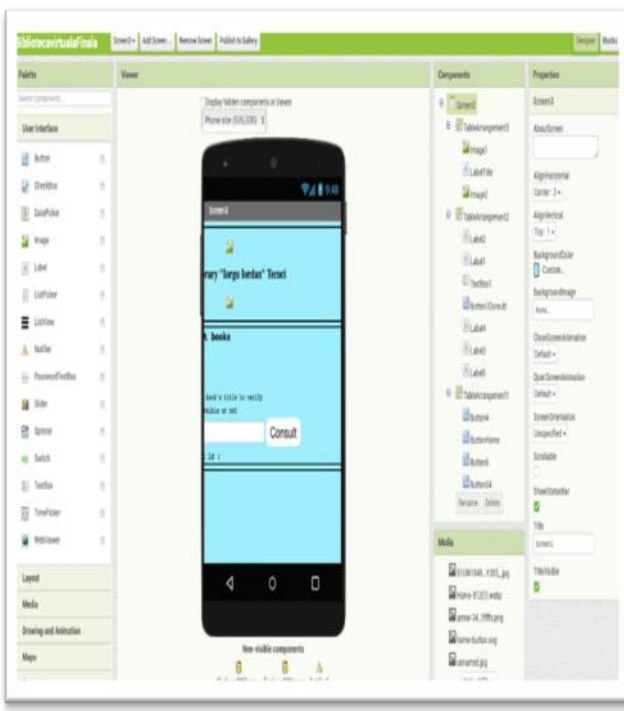
when ButtonAddBooks Click
do
  if
    TextBoxUsername Text == director and PasswordTextBox1 Text == 123
  then
    open another screen screenName Screen1
  else
    call NotificationCenter ShowAlert
    notice Wrong username or password
```



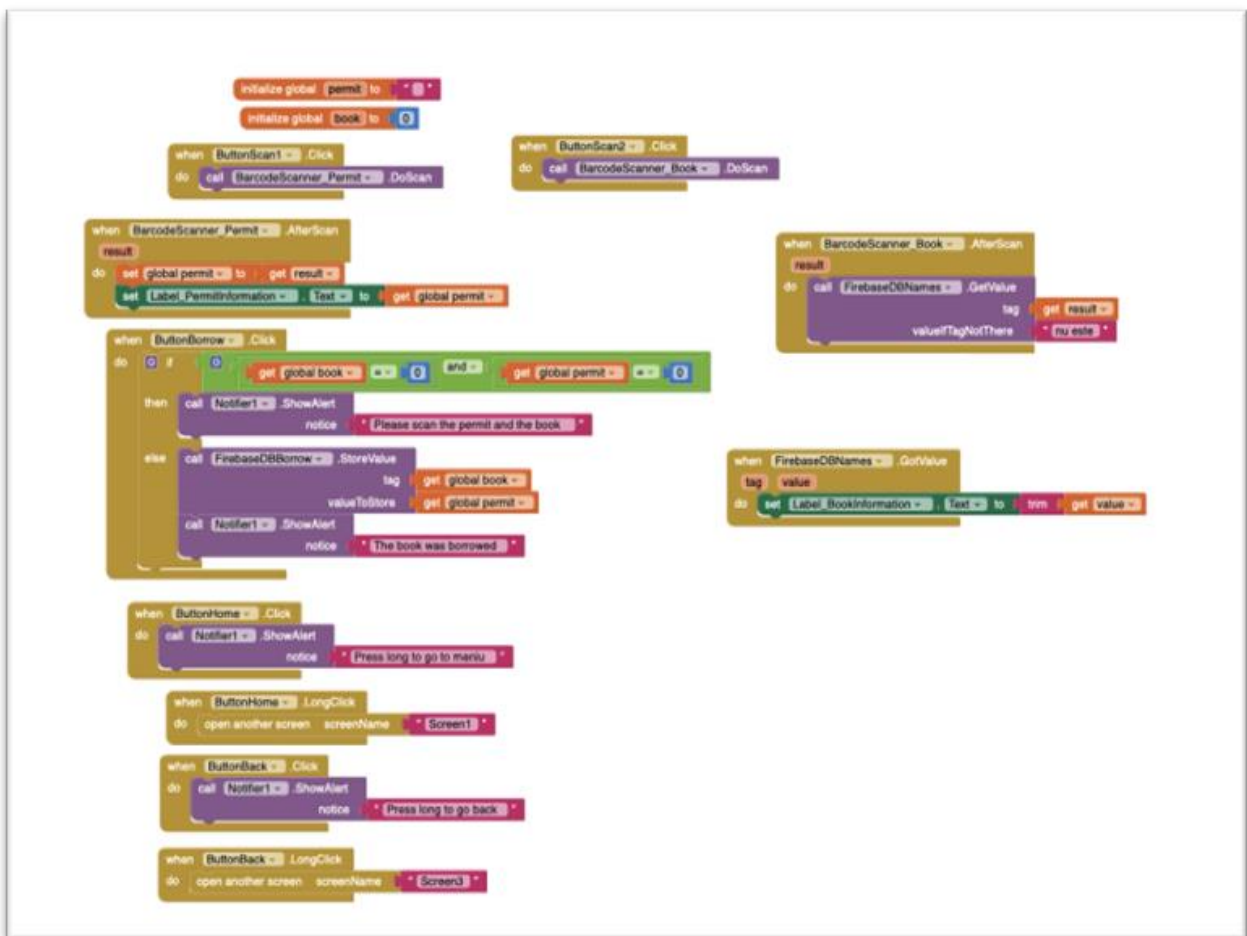
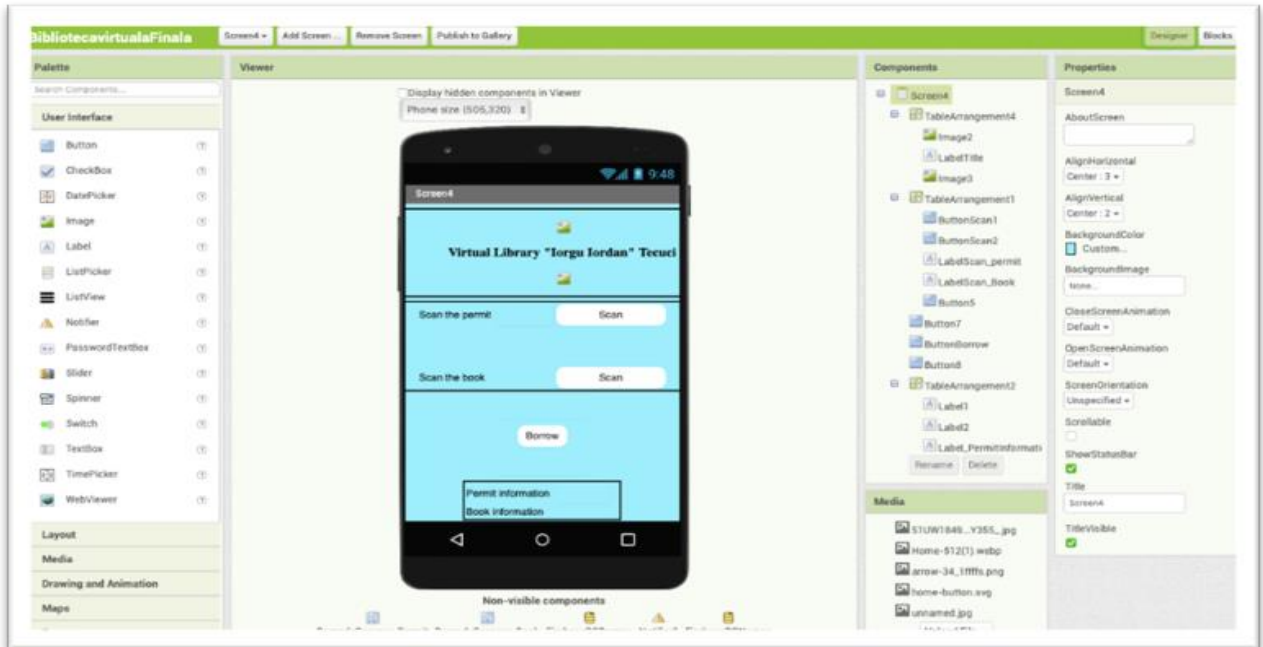
După cum se observă, în ecranul 2 se face **înregistrarea utilizatorilor**, printr-un formular de validare a numelui de utilizator și a parolei. Dacă datele introduse sunt valide, utilizatorul va fi înregistrat într-o bază de date internă, cu ajutorul TinyDB, element din platformă.



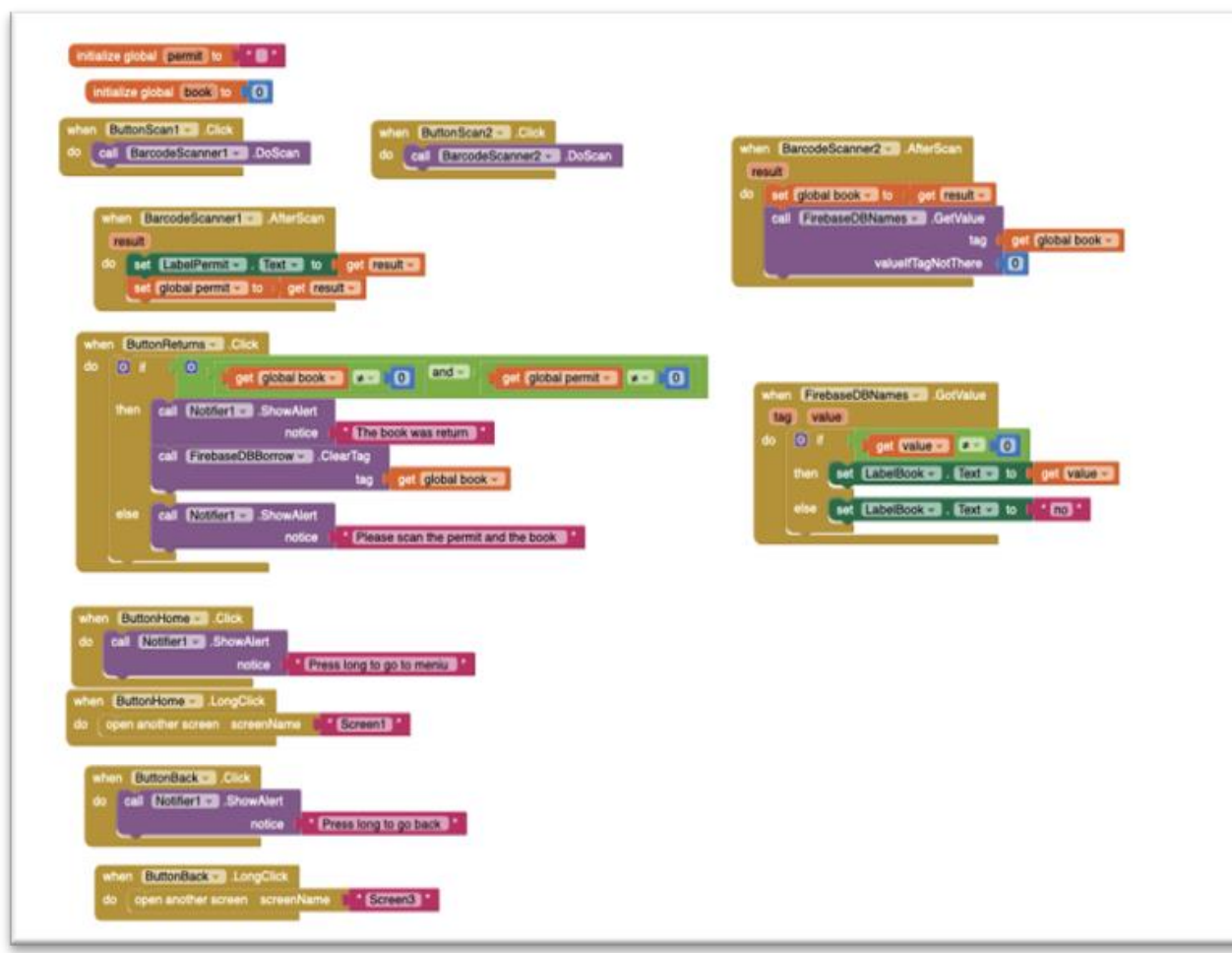
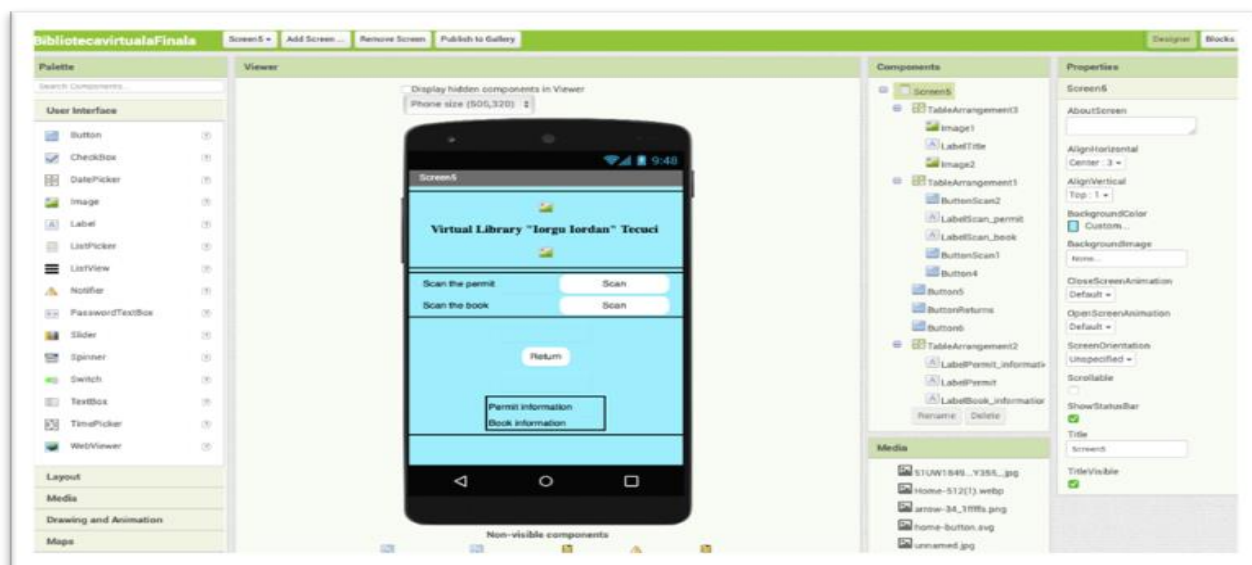
Ecranul 3 realizează **consultarea cărților în baza de date externă**, stocată pe situl firebase (aici se fac configurații suplimentare – cheia de autentificare și link-ul către baza de date)



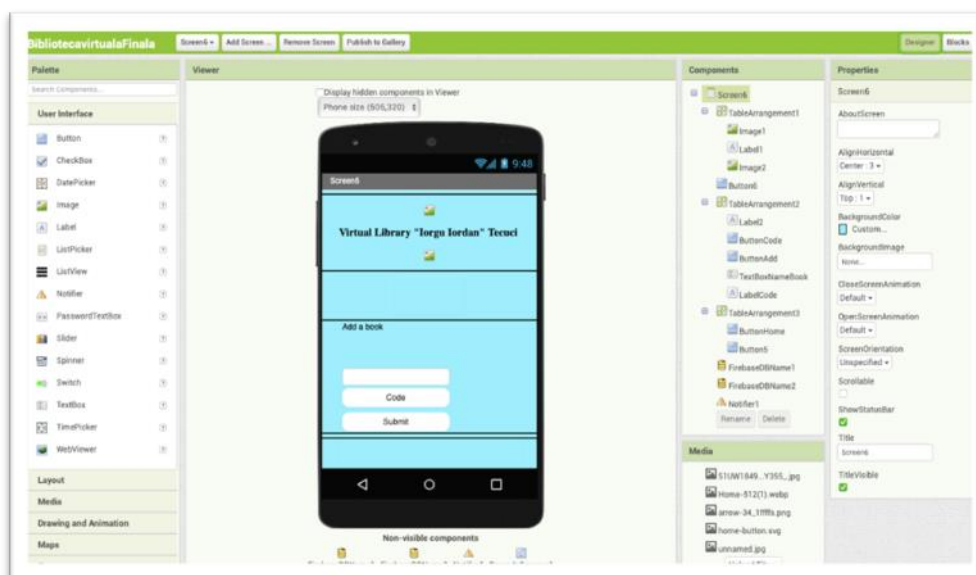
Ecranul al 4-lea este cel mai complex și *realizează împrumutul* efectiv prin scanarea succesivă a permisului, a cărții și apăsarea butonului de împrumut. Meniul este în engleză deoarece specificațiile concursului au fost de așa natură să comunicăm într-una din limbile franceză, engleză sau germană. Codul din spate este redat în poza următoare.



Ecranul al V-lea este similar cu cel de mai sus și realizează *restituirea cărților*. Pentru simplitate, la ideea domnului profesor, am folosit aceeași bază de date, restituirea reprezentând o adăugare în baza de date a cărții scanate iar împrumutul fiind o ștergere în baza de date a cărții împrumutate. Având în vedere timpul scurt pe care l-am avut la dispoziție, am considerat că astfel putem preveni o serie de anomalii de actualizare a bazei de date.



Ultimul ecran este *accesibil doar utilizatorilor cu rol de administrator, care pot edita, adăuga cărți în baza de date* astfel încât să existe un control asupra cărților din biblioteca virtuală.



În concluzie:

- * Aplicația poate fi îmbunătățită din punct de vedere grafic și funcțional, în realitate are o serie de imagini de fundal care nu sunt vizibile în momentul editării codului și în capturile de ecran incluse
- * Aplicația poate fi dezvoltată în sensul includerii unor funcții premium, cum ar fi accesul la cărți în format digital protejat
- * Aplicația este originală și extrem de utilă în momentul unei pandemii, a concediului sau lipsei bibliotecarului școlar
- * Se poate conecta cu alte aplicații, eventual de mesagerie, etc
- * Colaborarea în echipa **Codestar** ne-a adus un binemeritat premiu și ne-a descoperit noi pasiuni, ne-a stârnit curiozitatea spre aplicații mobile și credem noi, a legat o prietenie frumoasă între membrii echipei și mentorul nostru profesorul de informatică Cristi Rusu de la Școala Gimnazială „Iorgu Iordan” Tecuci.





```
// Includem librăriile necesare
#include "Arduino.h"
#include "SFE_BMP180.h"
#include "DHT.h"
#include "LDR.h"
#include "Wire.h"
#include "SPI.h"
#include "Adafruit_SSD1306.h"
#include "Adafruit_GFX.h"
// definim pinii utilizați
#define DHT_PIN_DATA 2
#define LDR_PIN_SIG A3
#define OLED128X64_PIN_RST 5
#define OLED128X64_PIN_DC 4
#define OLED128X64_PIN_CS 3
// definim variabilele globale și constantele
#define THRESHOLD_ldr 100
int ldrAverageLight;
// object initialization
SFE_BMP180 bmp180;
DHT dht(DHT_PIN_DATA);
LDR ldr(LDR_PIN_SIG);
#define SSD1306_LCDHEIGHT 64
Adafruit_SSD1306 oLed128x64
(OLED128X64_PIN_DC, OLED128X64_PIN_RST,
OLED128X64_PIN_CS);
// define vars for testing menu
const int timeout = 10000; //define timeout of 10
sec
char menuOption = 0;
long time0;
// Setup the essentials for your circuit to work. It runs
first every time your circuit is powered with electricity.
void setup()
{
// Setup Serial which is useful for debugging
// Use the Serial Monitor to view printed messages
Serial.begin(9600);
while (!Serial) ; // wait for serial port to connect.
Needed for native USB
Serial.println("start");
//Initialize I2C device
bmp180.begin();
dht.begin();
ldrAverageLight = ldr.readAverage();
oLed128x64.begin
(SSD1306_SWITCHCAPVCC); // by default, we'll
generate the high voltage from the 3.3v line internally!
(neat!)
oLed128x64.clearDisplay(); // Clear the buffer.
oLed128x64.display();
menuOption = menu();
}
// Main logic of your circuit. It defines the interaction
between the components you selected. After setup, it
runs over and over again, in an eternal loop.
void loop()
{
if(menuOption == '1') {
```



```

// BMP180 - Barometric Pressure, Temperature,
Altitude Sensor - Test Code

// Read Altitude from barometric sensor, note that
the sensor is 1m accurate

double bmp180Alt = bmp180.altitude();

double bmp180Pressure = bmp180.getPressure();

double      bmp180TempC      =
bmp180.getTemperatureC();    //See also
bmp180.getTemperatureF() for Fahrenheit

Serial.print(F("Altitude:  ")); Serial.print
(bmp180Alt,1); Serial.print(F(" [m]"));

Serial.print(F("\tpressure:  ")); Serial.print
(bmp180Pressure,1); Serial.print(F(" [hPa]"));

Serial.print(F("\tTemperature:  ")); Serial.print
(bmp180TempC,1); Serial.println(F(" [°C]"));
}

else if(menuOption == '2') {

// DHT22/11 Humidity and Temperature Sensor -
Test Code

// Reading humidity in %

float dhtHumidity = dht.readHumidity();

// Read temperature in Celsius, for Fahrenheit
use .readTempF()

float dhtTempC = dht.readTempC();

Serial.print(F("Humidity:  ")); Serial.print
(dhtHumidity); Serial.print(F(" [%]\t"));

Serial.print(F("Temp:  ")); Serial.print(dhtTempC);
Serial.println(F(" [C]"));

}

else if(menuOption == '3') {

// LDR (Mini Photocell) - Test Code

// Get current light reading, subtract the ambient
value to detect light changes

int ldrSample = ldr.read();

int ldrDiff = abs(ldrAverageLight - ldrSample);

Serial.print(F("Light Diff: ")); Serial.println(ldrDiff);
}

else if(menuOption == '4') {

// Monochrome 1.3 inch 128x64 OLED graphic
display - Test Code

oLed128x64.setTextSize(1);
oLed128x64.setTextColor(WHITE);
oLed128x64.setCursor(0, 10);

oLed128x64.clearDisplay();

oLed128x64.println("Circuito.io Rocks!");

oLed128x64.display();

delay(1);

oLed128x64.startscrollright(0x00, 0x0F);

delay(2000);

oLed128x64.stopscroll();

delay(1000);

oLed128x64.startscrollleft(0x00, 0x0F);

delay(2000);

oLed128x64.stopscroll();

}

if (millis() - time0 > timeout)
{
menuOption = menu();
}
}

// Menu function for selecting the components to be
tested

// Follow serial monitor for instructions

char menu()
{

Serial.println(F("\nWhich component would you

```



```

Serial.println(F("(menu) send anything else or press
on board reset button\n"));

while (!Serial.available());

// Read data from serial monitor if received
while (Serial.available())
{
  char c = Serial.read();
  if (isAlphaNumeric(c))
  {
    if(c == '1')
      Serial.println(F("Now Testing
BMP180 - Barometric Pressure, Temperature, Altitude
Sensor"));
    else if(c == '2')
      Serial.println(F("Now Testing
DHT22/11 Humidity and Temperature Sensor"));
    else if(c == '3')

```

```

Serial.println(F("Now Testing LDR
(Mini Photocell)"));
    else if(c == '4')
      Serial.println(F("Now Testing
Monochrome 1.3 inch 128x64 OLED graphic
display"));
    else
    {
      Serial.println(F("illegal input!"));
      return 0;
    }
    time0 = millis();
    return c;
  }
}
}

```

Arduino este o companie open-source care produce atât plăcuțe de dezvoltare bazate pe microcontrolere, cât și partea de software destinată funcționării și programării acestora. Pe lângă acestea include și o comunitate uriașă care se ocupă cu creația și distribuirea de proiecte care au ca scop crearea de dispozitive care pot sesiza și controla diverse activități sau procese în lumea reală.

Proiectul este bazat pe designul plăcilor cu microcontroler produse de câțiva furnizori, folosind diverse tipuri de microcontrolere. Aceste plăci pun la dispoziția utilizatorului pini I/O, digitali și analogici, care pot fi interfațați cu o gamă largă de plăcuțe numite scuturi (shield-uri) și/sau cu alte circuite. Plăcile au interfețe de comunicații seriale, inclusiv USB pe unele modele, pentru a încărca programe din calculatoarele personale. Pentru programarea microcontrolerelor, Arduino vine cu un mediu de dezvoltare integrat (IDE) bazat pe proiectul Processing, care include suport pentru limbaje de programare ca C și C++.

Primul Arduino a fost lansat în 2005, având ca țintă asigurarea unei soluții ieftine și simple pentru începători și profesioniști spre a crea dispozitive capabile să interacționeze cu mediul, folosind senzori și sisteme de acționare. Cele mai comune exemple sunt dispozitivele pentru utilizatorii începători precum: roboții simpli,



termostatele și/sau detectoarele de mișcare. Plăcuțele Arduino sunt disponibile comercial sub formă preasamblată sau sub forma unor kituri de asamblat acasă (do-it-yourself). Specificațiile schemelor sunt disponibile pentru orice utilizator, permițând oricui să fabrice plăcuțe Arduino. Adafruit Industries estimase la mijlocul anului 2011 că peste 300.000 de plăcuțe oficiale Arduino au fost produse, iar în 2013 700.000 de plăcuțe oficiale erau în posesia utilizatorilor.

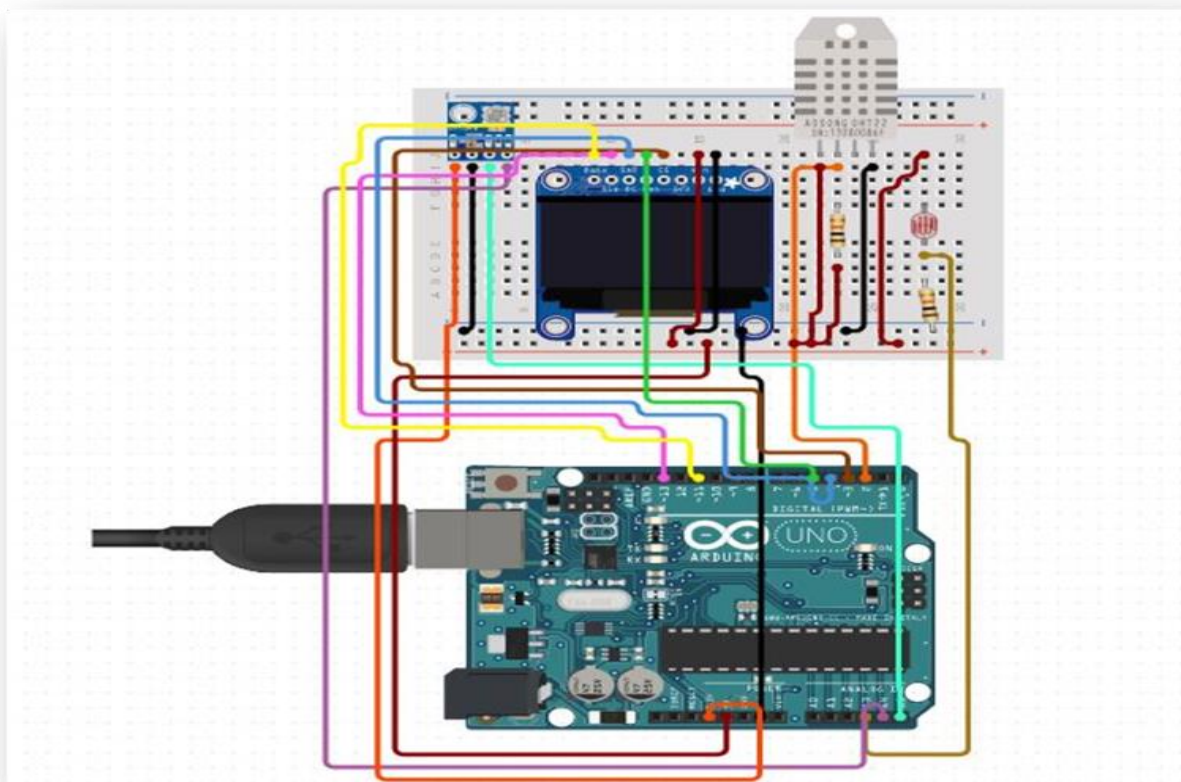
<https://ro.wikipedia.org/wiki/Arduino>



Cum să începi un club de programare

Ideea de a începe un club de programare în școală poate fi o provocare pentru oricine. La Școala Gimnazială Iorgu Iordan Tecuci, a început în februarie 2020 după ce cu 3 ani în urmă începea un curs opțional de introducere în robotică în care abordam teme din cele mai variate: de la noțiuni elementare de fizică (semiconductori, rezistențe, senzori) la noțiuni

descărcare și instalare pe orice sistem de operare, la adresa <https://www.arduino.cc>, va trebui să încercăm firmware-ul proiectului, în principal drivere necesare pentru buna funcționare a senzorilor. Structura de foldere este redată mai jos:



sumare de programare și înțelegea programelor realizate în C++ pentru a controla plăci Arduino și până la proiecte concrete a căror dificultate crește pe măsura acumulării cunoștințelor. Aplicațiile practice se desfășurau pe platforma www.circuito.io și decurge foarte ușor se alege tipul de plăcuță, se aleg senzorii iar platforma generează codul și firmware-ul necesar bunei funcționări. Pentru o stație meteo care are un senzor de umiditate și temperatură, unul de presiune, unul de lumină, un ecran și un ceas digital, circuitul arată ca în figura de mai jos. După ce am realizat circuitul, urmează partea de cod, prezentată în extenso într-un fișier separat. Înainte de a încărca acest cod, va trebui din programul Arduino IDE, disponibil pentru

Noutatea la momentul redactării articolului



Noutatea la momentul redactării articolului este că, s-a pus la punct și posibilitatea rulării online a proiectelor destinate plăcilor Arduino, chiar pe platforma producătorului, lucru care face posibilă executarea proiectelor și pe tabletele la care dreptul de a instala aplicații este restrâns. Acest lucru este evidențiat în imaginea de mai jos. Arduino IDE este disponibil pentru Windows, dar și pentru Linux sau MacOS.



Activitatea de programare la Școala Gimnazială Iorgu Jordan Tecuci a continuat cu roboți mai complecși, de tip follow-line mBot, OzoBot și cei de la NextlabTech, unde am câștigat un kit pentru școală.



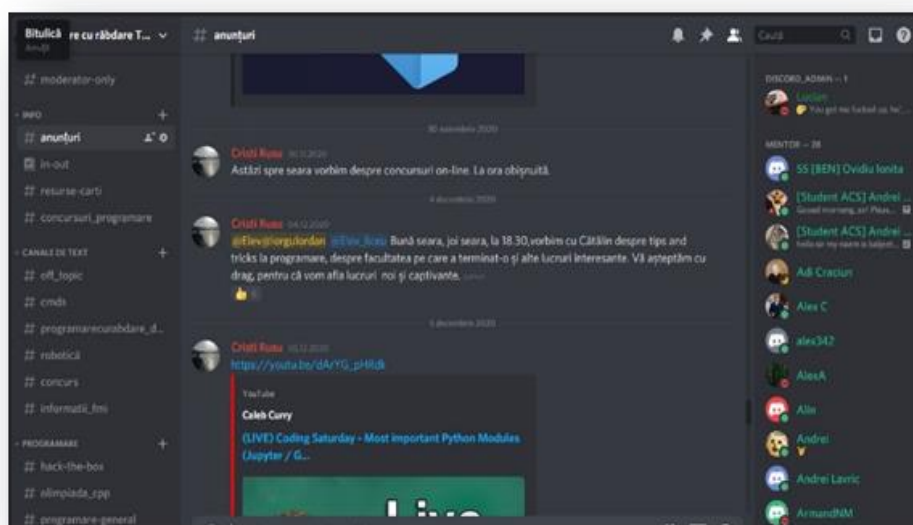


După aceste proiecte de robotică , ne-am alăturat inițiativei naționale Programare cu răbdare, devenind centrul local din Tecuci al acestui mare proiect, în februarie 2020. La nici o lună de la înființare, participam alături de Codevengers (Trifan Grigore – clasa a XI-a CNCH Tecuci, Andrășescu Alexandru – clasa a XI-a CNCH Tecuci, Aramă Vlad – clasa a XI-a CNCH Tecuci și Daraban Teodor – clasa a X-a CNAE Tecuci, elevi care în mai puțin de o lună au progresat cu Programare cu răbdare de la Școala Gimnazială Tecuci. Ei au utilizat o plăcuță Raspberry Py 3 ca server pentru aplicația lor, la Hackathonul Mărțișorului, unde am primit mențiune pentru un site realizat în Python folosind Flask și SQL Lite, un magazin virtual de mărțișoare.

După această competiție care a legat prietenii între cele 25 de echipe participante, ne-



am înscris la Empowersoft, secțiunea situri web unde am obținut din nou,



facultăților de profil, olimpici naționali la informatică, colaboratori de la Facultatea de Informatică Iași, ingineri angajați sau care au făcut stagii la Google, EA, Amazon, studenți de la universități de prestigiu naționale și internaționale, doctori în științe, din dorința de a-i motiva pe elevi să meargă mai departe, în aceste context dificil, să-și descopere pasiunea și să dea tot ce e mai bun. Am considerat că dacă măcar 1% din cei stârniți de curiozitate, vor urma acest drum spre industria IT, va fi un câștig pentru noi toți. Dintre subiectele abordate pe canalul de discord: programarea și designul jocurilor, inteligență artificială, Python, Java, aplicații mobile, admitere la FMI, FII, AC, algoritmică, POO, viața de programator și multe alte subiecte interesante.

Ce ne propunem? Să continuăm să scoatem la lumină nestemate, să le șlefuiim ca să lumineze și să inspire alți elevi în competițiile viitoare (2 echipe deja participă la un concurs de aplicații mobile – Hardcore Hackathon – și echipa de anul trecut continuă tradiția participând la Empowersoft, ediția de acest an.



Python este un limbaj de programare dinamic multi-paradigmă, creat în 1989 de programatorul olandez Guido van Rossum. Van Rossum este și în ziua de astăzi un lider al comunității de dezvoltatori de software care lucrează la perfecționarea limbajului Python și implementarea de bază a acestuia, CPython, scrisă în C.

Python pune accentul pe curățenia și simplitatea codului, iar sintaxa sa le permite dezvoltatorilor să exprime unele idei programatice într-o manieră mai clară și mai concisă decât în alte limbaje de programare ca C. În ceea ce privește paradigma de programare, Python poate servi ca limbaj pentru software de tipul *object-oriented*, dar permite și programarea imperativă, funcțională sau procedurală. Sistemul de tipizare este dinamic iar administrarea memoriei decurge automat prin intermediul unui serviciu „gunoier” (*garbage collector*). Alt avantaj al limbajului este existența unei ample biblioteci standard de metode. Implementarea de referință a Python este scrisă în C și poartă deci numele de *CPython*. Această implementare este software liber și este administrată de fundația *Python Software Foundation*.

Limbajele dinamice sunt deseori caracterizate prin tipizare dinamică, sunt interpretate, au management de memorie automatizat (*garbage collecting*) și au un nivel înalt de abstractizare. Tipul variabilelor nu este fixat la momentul declarării acestora (de fapt, declararea variabilelor nu este

necesară ca în C), ci este determinat de interpretator după conținutul lor sau după operațiile efectuate. Într-un asemenea caz se vorbește de tipizare dinamică.

Avantajul acestei metode de a lucra cu tipuri este că codul produs este mai clar și mai intuitiv. De exemplu:

```
a = 10 # a va fi considerat un număr întreg (integer)
b = 11.5 # b va fi considerat un număr rațional (double sau float)
```

```
c = "un rând de text" # c va fi considerat un șir (string)
```

În ciuda tipizării dinamice, Python este puternic tipizat, adică interpretatorul nu permite operații cu obiecte de tip diferit dacă acestea nu sunt bine definite, cum ar fi operațiile de tip cast în C. Alt concept important în Python este cel al tipurilor mutabile și nemutabile. Așa cum implică numele, datele cu un tip mutabil pot fi alterate după inițializare, în vreme ce la date nemutabile lucrul acesta este imposibil. Luând ca exemplu o listă (mutabilă), constatăm că putem adăuga elemente la listă sau le putem modifica fără probleme. În cazul unui șir de caractere (nemutabil), acest lucru nu este permis:

```
lista_mea = ["măr", "pară"]
lista_mea.append("strugure") # adaugă la listă
print lista_mea[2] # afișază: strugure
șirul_meu = "varză"
șirul_meu[0] = "b" # schimbă cuvântul în "barză" ->
Eroare
```

Sursa: <https://ro.wikipedia.org/wiki/Python>



Sondaje de opinie online. Aplicația Mentimeter.

www.mentimeter.com

Prof. Cubașa Paula, Liceul Teoretic SF. Maria, Galați

Odată cu tranziția recentă a multor domenii spre mediul online, educația a fost, la rândul ei, nevoită să se adapteze noilor cerințe pentru a continua să satisfacă nevoile elevilor și ale profesorilor. Predarea online presupune mult mai mult decât folosirea unui dispozitiv conectat la internet în locul unei săli de clasă. Este nevoie, de asemenea, de adaptarea modului de predare și de interacțiune cu elevii la noile cerințe. E greu a capta și menține atenția acestei generații a „digitalilor nativi”, a copiilor „născuți cu telefonul în mână”.

Aplicații pentru chestionare / sondaje de opinie

Aceste aplicații pot fi utilizate atunci când vă doriți să aflați părerea elevilor despre un anumit subiect, dar și atunci când doriți să verificați nivelul de cunoștințe. Aici avem câteva informații despre utilizarea aplicației Mentimeter și sunt prezentate facilități oferite de această platformă. Acest tip de aplicații poate fi folosit în diferite momente ale cursului. De exemplu, la începutul cursului, pot fi utile pentru a afla care sunt așteptările elevilor, care este nivelul actual de cunoștințe despre o anumită temă, ce concepte asociază cu un anumit cuvânt sau în ce măsură sunt de acord cu o afirmație, pe o scală de la 0 la 10, de exemplu. În timpul cursului, aplicațiile pot fi utilizate pentru a testa nivelul de cunoștințe al elevilor (prin adresarea de întrebări cu răspunsuri multiple). Totuși, în acest caz, cele două platforme sunt adecvate pentru un număr redus de întrebări (1-3); pentru a crea teste cu un număr ridicat de întrebări, sunt mai potrivite aplicațiile pentru evaluarea cunoștințelor. La finalul cursului, aceste instrumente digitale pot fi

răspunsuri deschise sau prin acordarea de note) sau pentru a le permite elevilor să adreseze întrebări.

MENTIMETER

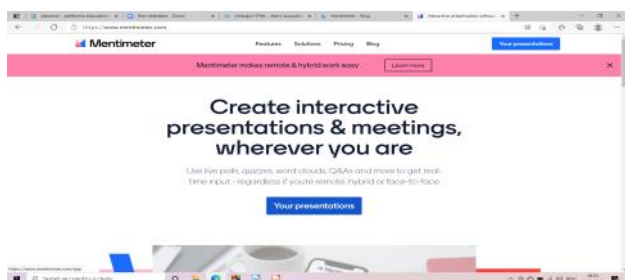
Mentimeter este o aplicație care îți permite să interacționezi cu un grup țintă în timp real. Este un instrument pentru sondaje unde poți pune întrebarea și grupul țintă poate da răspunsul folosind un telefon mobil sau orice alt dispozitiv conectat la internet. Să vedem cum funcționează!

Crearea unui sondaj

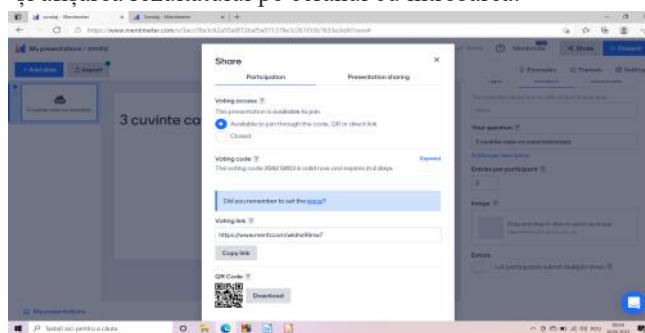


Pentru a crea un sondaj se va apăsa pe butonul "New presentation" (Prezentare nouă), apoi se va alege numele acestuia. În partea dreaptă va fi meniul pentru editarea tipului de întrebare. După această alegere veți pune titlul întrebării și opțiunile de răspuns și modul în care vor fi afișate.

Ultimul pas este prezentarea, ce se realizează cu ajutorul butonului "Present" din dreapta sus. După prezentarea întrebării în partea de sus a ecranului va fi vizibil un cod. Grupul țintă va accesa pagina aplicației și va introduce codul afișat în spațiul destinat acestuia, această acțiune fiind precedată de răspunderea întrebării pe dispozitivul propriu și afișarea rezultatului pe ecranul cu întrebarea.



folosite pentru a obține feedback cu privire la curs (prin



EVALUAREA ON-LINE - atractivitate și dinamism

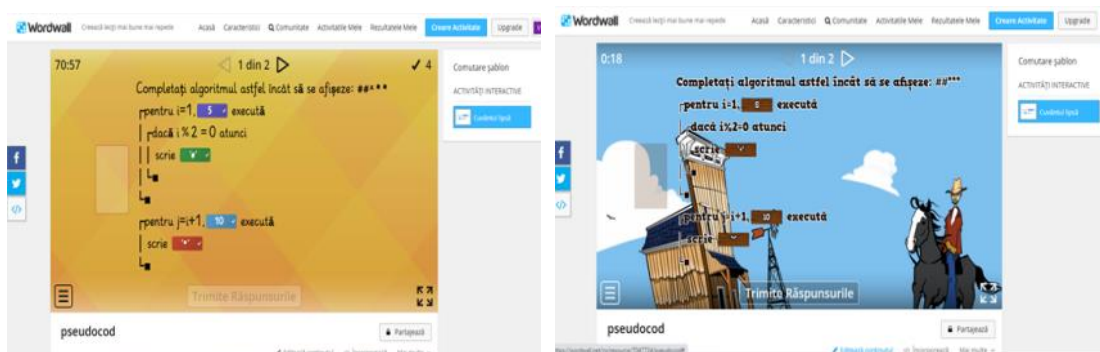
Prof. Tudor Veronica, Liceul Teoretic Dunărea

Dispozitivele informaționale și tehnologia vin în sprijinul stimulării imaginației elevilor astfel încât să faciliteze transferul achizițiilor învățării în viața de zi cu zi. Rolul profesorilor se schimbă, devenind creatori de material educativ, prin modelarea procesului de învățare: proiectează, implementează și evaluează lecțiile pe care le susțin, realizând designul instruirii.

Evaluarea interactivă nu constituie o noutate pentru profesorii de Informatică, care sunt familiarizați cu e-learningul și cu demersurile de realizare a activităților educative utilizând instrumente și resurse digitale. Totuși, evaluarea on-line înseamnă interactivitate, dinamism și adaptarea continuă a instrumentelor de evaluare la așteptările elevilor, la nivelul lor de cunoștințe, la noutățile din domeniu, astfel încât să avem ca finalitate formarea competențelor vizate.

Instrumentele de evaluare on-line ne oferă o gamă largă de posibilități de evaluare, o parte dintre aceste aplicații fiind foarte complexe. Una dintre aplicațiile on-line ce ne oferă multiple modalități de realizare a unor teste interactive este aplicația: wordwall. Wordwall este aplicația unde putem implementa diverse activități atractive pentru elevi: fișe de lucru, chestionare, chestionare concurs, identificarea unor cuvinte lipsă, deschide caseta și altele. Folosind Wordwall, putem crea cu ușurință un set de întrebări pentru etapa de evaluare dar și unele aplicații ce pot fi folosite în predarea sincronă sau asincronă. Astfel toți elevii vor fi implicați într-o mod interactiv în rezolvarea testului.

O primă facilitate oferită de aplicația Wordwall este aceea că ne pune la dispoziție mai multe teme predefinite pentru resursele create, astfel avem posibilitatea de a personaliza tema unui test în funcție de vârsta elevilor. Exemplul de mai jos implementează un test la capitolul Algoritmi elementari-pseudocod și poate fi aplicat la clasa a IX-a în varianta unei teme clasice, dar aceluiași test i se poate atașa o temă mai atractivă, testul putând fi astfel aplicat unei clase de gimnaziu.



Testele pot fi atribuite elevilor și permit modul de lucru multiplayer: aceeași resursă să fie folosită de mai mulți elevi în același timp, fiecare de pe dispozitivul propriu. Rezultatele individuale ale elevilor sunt disponibile după ce elevul a terminat activitatea. Wordwall poate fi utilizat pentru a crea activități atât interactive, cât și imprimabile. Cele imprimabile pot fi tipărite direct sau descărcate ca fișier PDF. Avem posibilitatea de a edita și tipări o resursă publică deja creată de către alți utilizatori sau să ne facem cont pe platformă și să creem materiale pe care să le stocăm în mod privat.

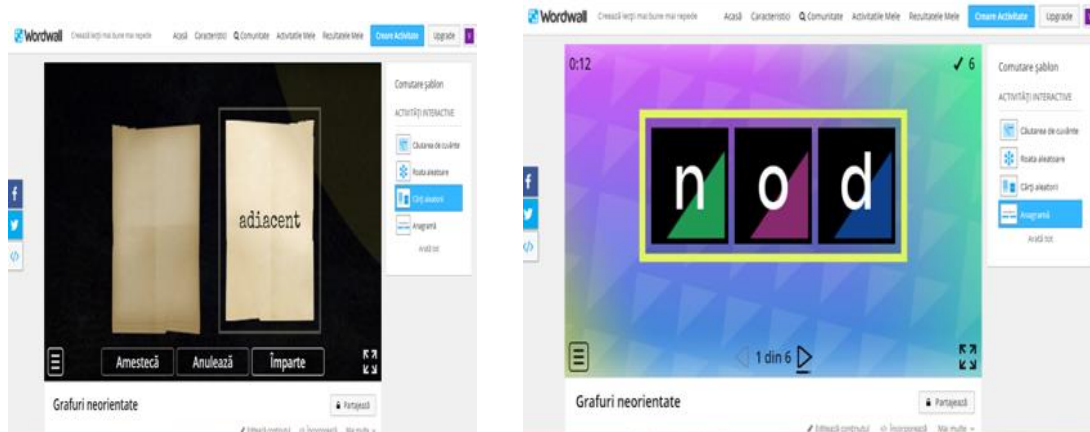
O altă facilități oferită de aplicația Wordwall este aceea că un test creat pe un anumit șablon poate fi comutat pe alte șabloane și avem astfel un alt format de test mai atractiv. Un astfel de test poate fi folosit la evaluare orală, la fixarea unor noțiuni teoretice, dar se poate aplica și ca feed-back la o lecție de predare a noilor cunoștințe. Exemplele de mai jos implementează un test la capitolul Grafuri neorientate-clasa a XI-a și folosesc șabloanele: roată aleatoare, căutarea de cuvinte, cărți aleatorii și anagrame. Cuvintele folosite în acest test sunt: *graf, nod, grad, lanț, conexitate și adiacență*.

Primul exemplu poate fi folosit la evaluarea orală și elevii trebuie să definească noțiunea selectată în urma învârtirii roții. Dacă modificăm șablonul în *căutare de cuvinte*, elevii trebuie să identifice cuvintele ce definesc noțiunile prezentate în lecție dintr-o grilă de caractere.



Același test poate fi afișat și în șablonul *cărți aleatorii* sau *anagramă*, în ultima variantă de șablon, elevii trebuie să amestece literele ce apar pe ecran astfel încât să obțină cuvântul ce definește una din noțiunile prezentate în lecție. Astfel un singur test poate fi utilizat în mai multe variante, folosind tiparele compatibile cu tipul de item ales.

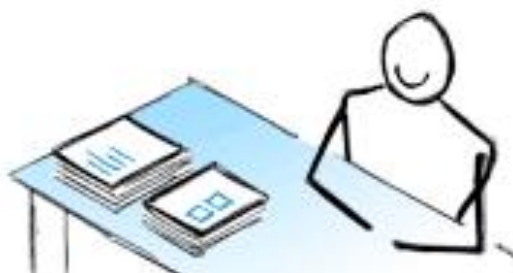




Un alt avantaj oferit de această aplicație este posibilitatea de a limita timpul de răspuns pe care elevii îl au la dispoziție pentru un item, dar și afișarea cronometrului, astfel încât elevii au posibilitatea să-și gestioneze timpul. Aceste teste interactive pot fi făcute publice astfel încât să fie folosite de către colegii profesori la ore sau putem opta ca testul interactiv creat de noi să nu fie vizibil pentru alți creatori de conținut educațional.

Instrumentele de evaluare on-line ne pun la dispoziție o gamă largă de posibilități de evaluare, uneori fiind mult mai multe variate de implementare decât suntem noi profesorii dispuși să explorăm. Pe lângă exemplele descrise mai sus, aplicația ne pune la dispoziție un set de standarde și specificații extrem de utile în evaluarea interactivă. În oricare din formele de evaluare prezentate anterior, în momentul în care stabilim structura testului de verificare trebuie să avem în vedere corelarea conținutului cu obiectivele învățării: cunoaștere, înțelegere, aplicare, analiză, sinteză, evaluare, aptitudini și deprinderi.

Profesorul trebuie să adapteze permanent strategiile moderne de evaluare astfel încât să ofere elevilor suficiente și variate posibilități de a demonstra ceea ce au învățat și mai ales să aplice cunoștințele acumulate. În acest sens evaluarea interactivă conduce la creșterea potențialului intelectual și creativ al elevilor prin solicitarea unui efort personal în actul învățării.



BitDefender-o companie de IT românească



Bitdefender este o companie românească producătoare de soluții de securitate cibernetică, cu activitate la nivel global. Compania a fost fondată și este deținută de *Florin Talpeș*, unul dintre cei mai cunoscuți antreprenori IT din România. În 2019 tehnologiile Bitdefender sunt prezente în 38% dintre soluțiile de securitate de pe piața internațională. Începând cu noiembrie 2011, există și o versiune gratuită, disponibilă pentru utilizatorii din România.

Cu sediul central în București, România, Bitdefender are reprezentanțe în SUA, Germania, Spania, Emiratele Arabe Unite și Marea Britanie. La începutul lui 2018 compania avea reprezentanțe în mai mult de 100 de țări.

Performanța tehnologiei Bitdefender a fost măsurată folosind rata de detecție a virusilor noi și necunoscuți într-un test realizat de către institutul de testare Anti Malware. Bitdefender a obținut, de asemenea, o certificare avansată de la AV Comparatives pentru scanarea la cerere și certificare la nivel standard pentru detectarea virusilor necunoscuți, deși acesta a fost criticat pentru viteza de scanare mai lentă.

B-HAVE

În mai 2006 Bitdefender a introdus o nouă tehnologie, **B-HAVE**, pentru a reduce dependența de semnături de virusi prin detecția interactivă a amenințărilor necunoscute. Această tehnologie se bazează pe analiza comportamentală într-un mediu virtualizat.

Pentru a determina dacă un program este periculos sau nu, această tehnologie simulează un computer virtual în care fișierele sunt executate și analizate. Pentru a simula configurația unui PC real, acesta emulează și dispozitive hardware virtuale.

Un program este considerat periculos în cazul în care încearcă să modifice anumite fișiere, să citească sau să scrie într-o zonă sensibilă de memorie sau să creeze un fișier care este asociat unui virus cunoscut.

Când întâlnește un program suspect, B-HAVE întârzie lansarea programului, iar caracteristicile sunt analizate și evaluate în mediul virtual.

Când sunt detectate acțiuni periculoase, B-HAVE pornește programul în mod normal. Dacă detectează un comportament suspect, B-HAVE trece fișierul în carantină sau șterge în mod automat aplicația suspectă.

Active Virus Control

În august 2009, Bitdefender introduce o nouă tehnologie, numită Active Virus Control, integrată în nouă gamă de produse.

Aceasta monitorizează continuu procesele specifice fiecărui program ce rulează în timp real pe PC și reacționează la potențiale acțiuni care pot avea un impact negativ asupra sistemului.

Spre deosebire de tehnologiile euristice, care verifică fișierele executabile atunci când sunt accesate, Active Virus Control monitorizează tot ce fac aplicațiile pe toată durata execuției lor.

Paula Adriana Porumb

clasa a VI-a

Liceul Teoretic „Dunărea”





Facultatea de Automatică, Calculatoare, Inginerie Electrică și Electronică din cadrul Universității “Dunărea de Jos” din Galați, Colegiul Național „Alexandru Ioan Cuza” și Casa Corpului Didactic Galați au organizat, în perioada 12-14 mai, un concurs de creativitate în domeniul IT, destinat celor pasionați de acest domeniu. Competiția s-a numit Severin Bumbaru, după numele unui mare mentor și creator, reprezentativ învățământului gălățean, în domeniul informaticii. „Concursul de creativitate în IT Severin Bumbaru” a avut ca scop atât stimularea creativității, originalității și gândirii critice, cât și aplicarea cunoștințelor acumulate din domeniul IT.

Ediția de anul acesta s-a desfășurat ONLINE în format hackathon de 2 zile, având o tematică propusă. Secțiunea “Animații Scratch” a atras aproape cincizeci de elevi din învățământul gimnazial. Fiecare participant a avut la dispoziție 15 de minute pentru prezentarea animației și încă 10 minute de conversație cu juriul. Elevii s-au întrecut în proiecte interesante și originale, arătându-și pasiunea pentru domeniul informatic încă de la clasele a V-a.

Au participat elevi atât din Galați cât și din Tecuci, Brăila și Pitești. Participanții au fost încântați de premiile consistente, oferite de numeroși sponsori generoși. [Anul viitor se va organiza o nouă ediție a concursului care sperăm să se desfășoare în condiții de normalitate, cu un număr și mai mare de participanți.](#)

Mulțumim Thecon, Cleste.ro, Logiscool Galați, Centrico Selir, Crystal System, Citadel HUB, R Systems Europe, Intel, StillCo IT&C Specialists pentru prezența dumneavoastră pe parcursul acestui eveniment!

Câștigătorii de la Secțiunea Animații Scratch sunt:

PREMIUL SPECIAL PENTRU CEA MAI REUȘITĂ ANIMAȚIE se acordă lui Voinea Ioan, elev clasa a 7-a la Liceul Pedagogic D. P. Perpessicius, Brăila cu punctajul maxim 100 pc.

*** CLASA a 5-a**

Premiul 1: Ciucă Alexia Gabriela, de la Colegiul Național Vasile Alecsandri cu un punctaj de 97 pc.

Premiul 2: Condrici Mihai, de la Școala Gimnazială Nr.29 cu un punctaj de 91 pc.

Premiul 3: Drugan Darius Ștefan, de la Colegiul Național "Al. I. Cuza" cu un punctaj de 82 pc.

Mențiunea 1: Știubianu Pavel Alexandru, de la Colegiul Național "Al. I. Cuza cu un punctaj de 77 pc.

Mențiunea 2: Cristea Tudor Andrei, de la Școala Gimnazială Miron Costin cu un punctaj de 73 pc.

*** CLASA a 6-a**

Premiul 1: Miron Eric Andrei, de la Colegiul Național "Al. I. Cuza" cu un punctaj de 95 pc.

Premiul 2: Lăcătuș Dragoș, de la Școala Gimnazială Nr. 28 cu un punctaj de 91 pc.

Premiul 3: Popescu Delia Ioana, de la Școala Gimnazială Nr.7 cu un punctaj de 88 pc.

Mențiunea 1: Zgorcea Mihai Alexandru, de la Școala Gimnazială Mircea Eliade Pitești cu un punctaj de 84 pc.

Mențiunea 2: Radu Andrei, de la Liceul Teoretic Emil Racoviță cu un punctaj de 82 pc.

Mențiunea 3: Baci Roberto Denis, de la Școala Gimnazială Nr. 7 cu un punctaj de 80 pc.

*** CLASA a 7-a**

Premiul 1: Bostan Cristian Ionuț, de la Colegiul Național Calistrat Hogaș cu un punctaj de 98 pc.

Premiul 2: Condrache Răzvan Ștefan, de la Colegiul Național Calistrat Hogaș cu un punctaj de 92 pc.

Premiul 3: Drăgan Paula Beatrice, de la Colegiul Național Calistrat Hogaș cu un punctaj de 90 pc.

Mențiunea 1: Stanciu Mihail, de la Școala Gimnazială Iorgu Iordan Tecuci cu un punctaj de 86 pc.

Mențiunea 2: Seria David, de la Școala Gimnazială Nr.7 cu un punctaj de 82 pc.

*** CLASA a 8-a**

Premiul 1: Sandu Robert, de la Școala Gimnazială "Ștefan cel Mare" cu un punctaj de 74 pc.

Premiul 2: Di Grandi Mario, de la Școala Gimnazială Nr. 29 cu un punctaj de 70 pc.

Felicitari și mult succes tuturor participanților în proiectele viitoare! Anul viitor se va organiza un nou concurs care sperăm să se desfășoare în condiții de normalitate, cu un număr și mai mare de participanți!



Cum filmezi desktop-ul?

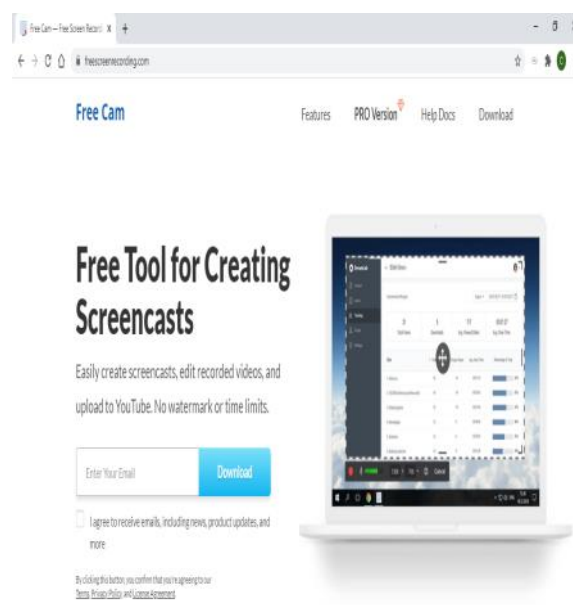
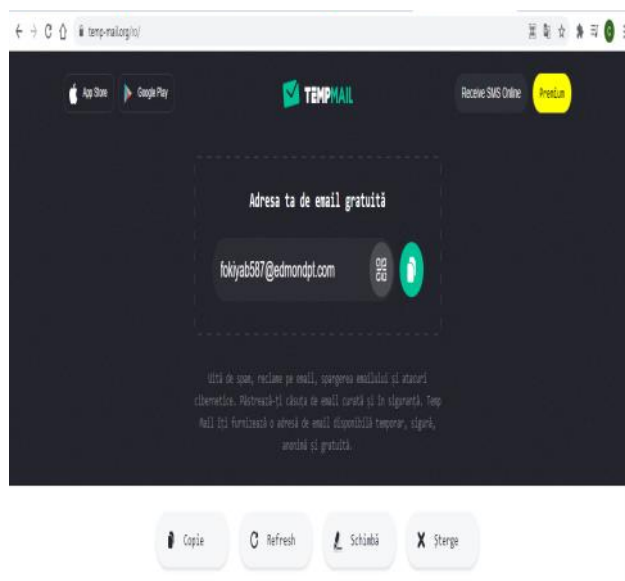
Prof Cristina Oprea
Liceul Teoretic "Emil Racoviță"

Cele mai multe dintre programele foarte cunoscute pentru filmarea desktop-ului (OBS, Action Mirillis, Bandicam, Fraps) pot avea consum foarte mare de resurse sau pot aplica watermark filmărilor create, fiind necesară achiziționarea versiunii premium.

FreeCam este un program mai puțin cunoscut.

Link-ul de unde se poate descărca programul:

<https://www.freescreeenrecording.com/>

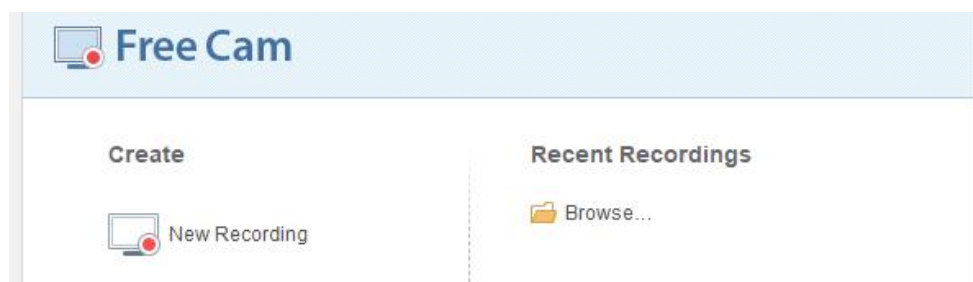


În câmpul în care se cere adresa de email, în cazul în care nu dorim să introducem adresa proprie, putem genera o adresă de email de unică folosință (adresă temporară care se auto-distruge după un anumit timp). Astfel, putem evita spam-urile în propria adresă de email. Un serviciu care generează adrese de email temporare este:

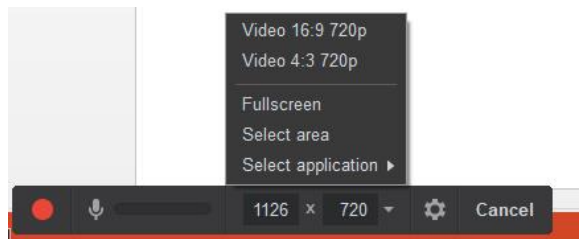
<https://temp-mail.org/ro/>



Dupa descărcarea, respectiv instalarea aplicației **FreeCam**, accesând butonul *Launch*, fereastra de înregistrare va fi deschisă.

Pentru a începe o filmare, click pe *New Recording*.

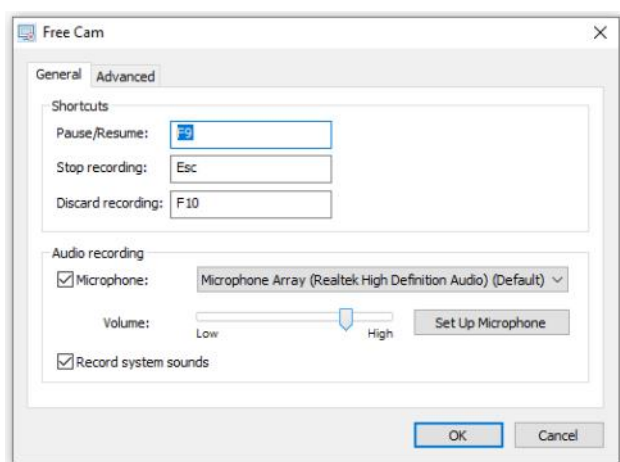


Se pot face diferite setări. În primul rând trebuie stabilită o rezoluție. Dacă dorim filmare în HD vom completa valorile 1280 x 720. Dacă dorim să filmăm full screen vom alege opțiunea corespunzătoare.

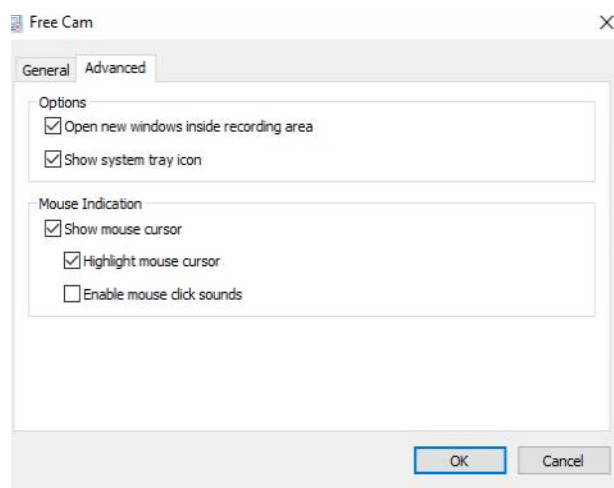


Accesând opțiunea *Setări*  putem stabili câteva shortcut-uri. Filmarea pornește dând clic pe butonul .

Microfonul se folosește dacă dorim înregistrarea cu microfon. Pentru aceasta, trebuie bifată caseta *Microphone* și selectat microfonul conectat la calculator.

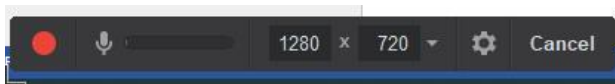


În caz că se aude cu puțin zgomot de fundal, mutăm cursorul volumului mai spre stânga. Dacă nu dorim să se audă sunetele sistemului, debifăm caseta *Record system sounds*.



În tab-ul *Advanced* se găsesc câteva setări importante, cum ar fi cele legate de cursor.

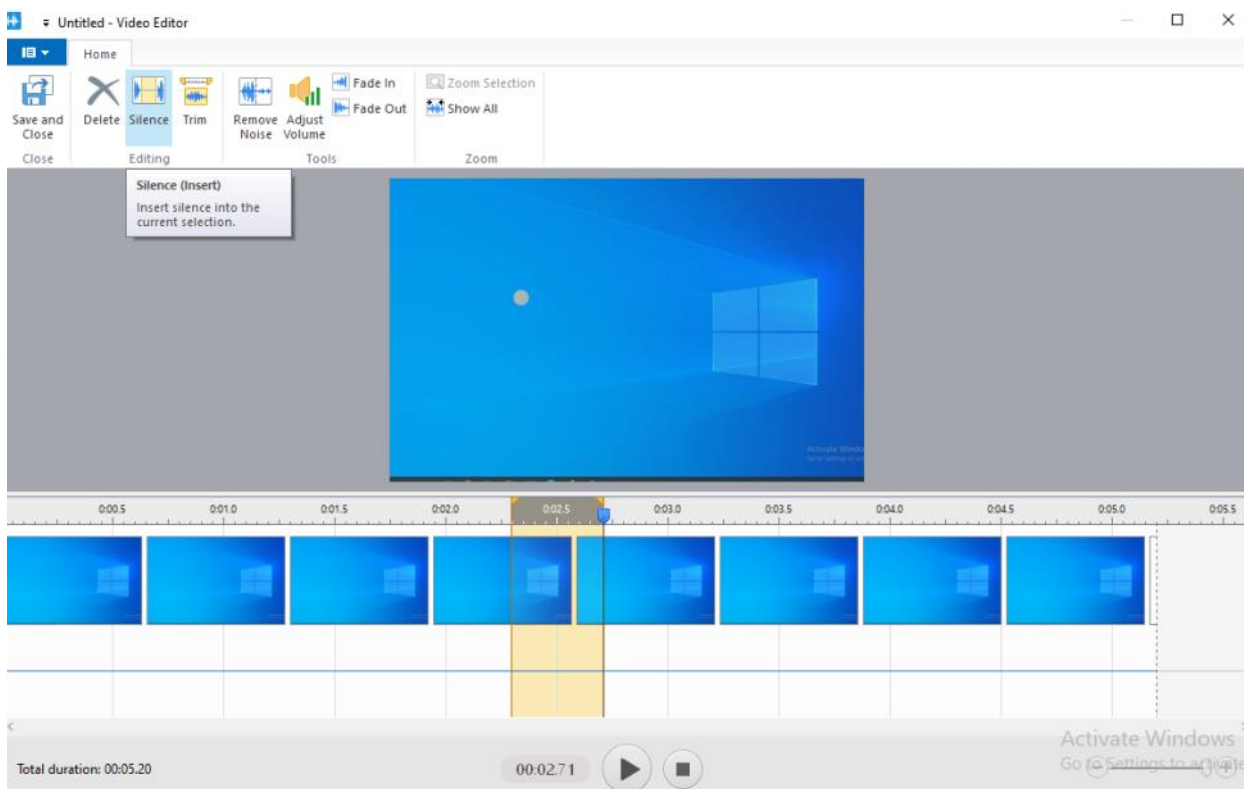
Dacă dorim ca acesta să se vadă pe filmare trebuie bifată caseta *Show mouse cursor*, iar dacă dorim să îl și evidențiem printr-un cerc galben în jurul său, bifăm și caseta *Highlight mouse cursor*. În timpul înregistrării microfonul poate fi ținut fie închis, fie deschis. (Putem realiza înregistrări numai pe baza indicațiilor prin intermediul ecranului, fără explicații verbale.)



După realizarea unei filmări putem previzualiza rezultatul și, totodată, putem edita anumite secvențe.

Nu se poate realiza această operațiune ca într-un program profesional de editare, dar programul oferă câteva unelte de editare.





După ce sunt făcute aceste setări se dă click pe *Save and Close* și totul va fi salvat cu modificările realizate.

Videoclipul poate fi postat direct pe *youtube*. Poate fi și salvat în format WMV (Windows Media Video) dar, dacă dorim, îl putem converti în alt format dorit (de exemplu, MP4).



BLOCAJE ÎN COMUNICAREA DIDACTICĂ ÎN MEDIUL ONLINE

Ing. Damian Felicia-Anișoara
Liceul Teoretic „Emil Racoviță” Galați

În fața pandemiei globale actuale, elevii din întreaga lume nu au avut de ales decât să-și migreze mediile de predare către platforme instituționale online, precum și pe platforme gratuite (precum ZOOM, Discord, tlk.io, Google Hangouts sau YouTube, Skype etc.).

În timp ce elevii se adaptează la un nou mod de predare, elevii au trebuit să se adapteze la un nou mod de învățare. În contextul actual, ne întrebăm:

învățarea online schimbă elevii și profesia didactică ?
care este impactul predării online asupra identității profesionale pentru elevi care au trecut la clasa online?

Dobândirea unor abilități pedagogice și tehnice adecvate sunt două elemente cruciale pentru trecerea cu succes la clasa online. Succesul orelor și al programelor online depinde în mare măsură de utilizarea centrării pe elev. Astfel, rolul elevului online trebuie să proiecteze, să creeze și să faciliteze interacțiuni bogate între elevi pentru a-i ține motivați. În orele online bine concepute, elevilor li se cere frecvent să accepte responsabilități suplimentare, dintre care unele erau odinioară apanajul elevilor. Pe lângă provocarea pedagogică a adaptării la mediile de învățare online și

indiferent de asistența tehnică pe care o pot primi de la școală, elevii trebuie să fie

competenți în utilizarea tehnologiilor educației la distanță pentru a putea selecta instrumentele care vor le permite să-și îndeplinească obiectivele de instruire. Această cerință poate reprezenta o semnificativă provocare pentru elevii care au intrat în profesie într-un moment în care expertiza tehnologică nu era necesar.

În timp ce dobândirea unor competențe tehnice și didactice adecvate este esențială, nu este un

garanția succesului. Învățarea online este adesea la fel de nouă pentru elevi ca și pentru profesori. Și elevii trebuie să facă ajustări la obiceiurile lor de studiu pentru a avea succes online. S-a raportat lipsa disciplinei elevilor

ca o barieră principală în calea învățării online. Există și alte bariere implicate în învățarea online:

- **Camera web** - lupta dintre prezență și absență

O problemă importantă care afectează interacțiunea profesor-elev este vizibilitatea arătată de elevi în timpul orelor. Unii dintre ei preferă să închidă camerele web și, făcând acest lucru, construiesc un zid între ei și profesori și, de asemenea, între ei și colegii lor. În această situație, elevul nu poate primi feedback complet de la colegii săi, sau de la profesor ceea ce crează dificultăți în angajarea la alte discuții și, uneori, profesorul ar putea simți că vorbește singur. De asemenea, dacă camera web este oprită, acest lucru ar putea însemna că nu mai sunteți acolo, ați părăsit sala de clasă pentru a face altceva. Aceste probleme ar putea fi rezolvate prin includerea în program a unei condiții care impune tuturor elevilor să păstreze camera web pornită pentru a asigura participarea la clasă și încrederea reciprocă între profesor și elev.

- **Barierile sonore.** Una dintre problemele apărute în contextul învățării virtuale prin intermediul platformelor de predare video este interferența audio cauzată de zgomotul de fond. Alte interferențe de zgomot sunt cauzate de activitățile simultane pe care participanții la apel (elevi sau profesori) le desfășoară în timpul cursului (cum ar fi mâncarea). Interferențele de zgomot pot fi cauzate și de ceilalți membri ai familiei care își desfășoară activitățile (participarea la cursuri sau activități de lucru) de acasă. Astfel, o coordonare slabă cu alți membri ai familiei poate perturba participarea la ore. Aceste probleme de zgomot pot fi rezolvate prin închiderea microfonului personal atunci când profesorul sau altcineva vorbește. În același timp, coordonarea prealabilă cu membrii familiei sau cu alți colegi poate ajuta la rezolvarea obstacolelor nedorite ale zgomotului și poate facilita concentrarea în timpul orelor de curs.



<p>- Abilități tehnice și digitale slabe Alte probleme care ar putea apărea pot fi legate de lipsa abilităților digitale în ceea ce privește utilizarea platformelor de către profesori și elevi. Astfel, ar putea exista cazuri în care elevii și profesorii nu pot utiliza în mod corespunzător platformele educaționale virtuale. O posibilă soluție pentru rezolvarea acestor probleme ar putea fi implementarea unor ghiduri de utilizare online sau asistență tehnică.</p> <p>Amintim că, potrivit regulilor de desfășurare a școlii online și școlii hibrid, elevii au avut obligația de a participa la ore, dar aceleași reguli nu precizează detalii precum folosirea camerei video. Dacă unii elevi au impus elevilor din clasă să țină camera pornită, așa încât “să fie văzuți”, alții nu au reușit acest lucru, mai ales în cazul elevilor mari, care au invocat aspecte legate de viața personală, aspect fizic, mediul de acasă. Așa pot apărea situații în care elevii par mai degrabă să se joace sau să facă alte activități, în spatele “ecranului negru”, sau, la elevi și colegi, senzația de “oboseală de Zoom”, produsă de perturbările care apar când comunicarea normală (predarea la clasă, ședințele) e înlocuită cu astfel de comunicări la distanță – cu sau fără video.</p> <p>Vorbind dintr-un mediu în care închiderea camerei video de către elevi este mai degrabă excepția decât regula, Katie Seltzer, absolventă de Harvard și profesoară de liceu din Portland, SUA, oferă pentru portalul educațional Edutopia câteva sfaturi în această privință. Ea pleacă de la ideea că, dacă elevii au opțiunea să își închidă camerele video în timpul orelor, ei s-ar simți mai în largul lor, într-un context și așa stresant, iar aceasta ar deschide noi posibilități de predare.</p> <p>Ea amintește că, în mod normal, apela la o soluție clasică atunci când dorea, la clasă, să evalueze gradul de implicare al elevilor în ora de curs: se uita la fețele lor. Acest lucru – a privi chipuri – este foarte posibil în școala online, dacă elevul ține camera video pornită, dar schimbarea mediului produce schimbări de comportament invocate ca surse de oboseală de Zoom, de la tendința de a te uita fix la propria imagine, la senzația că poți face multe alte lucruri între timp, fără să fie neapărat așa, sau la înclinația de a privi camera pentru a sugera că ești atent la discuție, când, în condiții normale, la ore sau ședințe atenția nu se manifestă în mod natural astfel.</p> <p>Propunerile prezentate de Edutopia pentru creșterea par-</p>	<p>ticipării la ore a elevilor cu video închis:</p> <ol style="list-style-type: none"> 1. Includerea lor în grupuri restrânse de discuție ale elevilor, pentru lucrul la proiecte. 2. Participanții nu sunt obligați să stea tot timpul cu camera pornită, dar trebuie să trimită, la final, înregistrări video din care să reiasă contribuția fiecăruia. 3. Aici pot apărea diverse alternative – autoarea dă exemplul unui exercițiu în care o parte dintre elevii din grupul de discuții închide camerele, alții nu, urmând să vorbească doar cei care au camera video pornită. Cei care au camerele închise trebuie să comenteze în programul de tip chat folosit. 4. Pentru momentele în care elevii închid camera, ei pot fi puși să contribuie la discuție atât audio, cât și prin comentarii text în serviciul de chat. 5. Ora poate fi structurată astfel încât elevii să poată să ia “pauze” în care să poată să închidă camera – de exemplu, pentru a lucra pe cont propriu sau în grupuri restrânse, la un proiect – urmând să revină la oră cu camera pornită. 6. Pot fi folosite, la oră, instrumente interactive care nu impun prezența “video” a elevului, instrumente ce pot fi folosite atât pentru creșterea nivelului de implicare, cât și în evaluare – sondajele sunt cel mai la îndemână exemplu. 7. Accentul pe instrumentele de tip “tablă de scris”, pentru a verifica nivelul de înțelegere a materiei predate. <p>Iar, la urma urmei, elevul poate să îl întrebe, constructiv și încurajator, pe elevul care închide camera video: “am văzut că ai închis video. Cum pot să te ajut să fii prezent în această comunitate online?”</p> <p>Recomandări pentru tipul de învățare la distanță:</p> <ul style="list-style-type: none"> → Asigurarea unei comunicări clare, nuanțate, explicite și, în același timp, ferme în cazul în care elevii adresează întrebări și omiterea unui așa-zis ”dialog suprapus” astfel încât, într-un timp limitat elevii să poată oferi clarificări specifice obiectivelor de învățare, cerințe-sarcină; → Monitorizarea continuă și activă a e-mail-ului de către fiecare cadru didactic, a platformei utilizate, chiar a telefonului mobil în vederea asigurării unei comunicări eficiente, și optime, de suport, cu toți părinții elevilor implicați în acest proces de învățare; → Evitarea, cu excepția cazului în care sunt programate cu atenție și limitate, conversațiilor on-line îndelungi, fără eficiență didactică sau afectivă;
--	--



→ Evitarea solicitării imprimării (listării) tuturor fișelor de lucru individual recomandate (Toate activitățile ar putea să fie finalizate pe un dispozitiv sau încărcate ca imagine);

→ Asigurarea și respectarea sistematică a activităților de pregătire teoretică și metodică individuală (sau de grup) a cadrelor didactice pe tot parcursul perioadei în care unitățile școlare sunt închise.

În concluzie consider că noile tehnologia informației va influența educația mai mult decât a făcut-o până în prezent, viitoarele programe de învățământ vor trebui să conțină instrumente de e-learning, orele se vor desfășura din ce în ce mai mult virtual, iar testarea online va devini

banală. Situația actuală a pandemiei a forțat sistemul educațional să se adapteze la predare prin intermediul platformelor virtuale și al soluțiilor digitale.

Bibliografie:

1. Bedenlier, S.; Wunder, I.; Gläser-Zikuda, M.; Kammerl, R.; Kopp, B.; Ziegler, A.; Händel, M. Generation Invisible. Higher Education Students' (Non) Use of Webcams in Synchronous Online Learning. Available online: https://webcache.googleusercontent.com/search?q=cache:aAOI_k7nAYkJ:https://psyarxiv.com/7brp6/download+&ccd=2&hl=en&ct=clnk&gl=ro



Tehnologia este o parte importantă pentru dezvoltarea personală a tinerilor secolului 21. Atunci când copiii fac cunoștință pentru prima dată cu programarea, încep să aprecieze mai mult tehnologia și modul în care funcționează dispozitivele. Iar atunci când aceștia reușesc să construiască propriul lor joc sau să programeze singuri un robot, satisfacția este imensă. A învăța programare este la fel de important ca a învăța matematică sau biologie. Este parte din viața noastră, societatea de astăzi bazându-se într-un procent covârșitor pe tehnologie.

Bill Gates spunea că a învăța programare îți deschide mintea, te ajută să gândești mai structurat și creează un mod simplu de a înțelege lucrurile, care poate fi aplicat și în alte domenii. A învăța despre cum funcționează programele sau despre cum sunt programate dispozitivele din jurul nostru, este doar un avantaj pe care programarea îl are asupra dezvoltării copiilor. Multe dintre avantajele programării sunt legate de partea cognitivă:

Atunci când învață să scrie cod, copiii își dezvoltă foarte mult gândirea computațională. Ei învață cum să rezolve o problemă prin organizarea logică a datelor, prin analiza acestora, prin identificarea și crearea de pași, prin "spargerea" problemei în părți mai mici. De asemenea, învățând programare, copiii își dezvoltă abilitățile de a recunoaște modele, de a folosi moduri abstracte în a rezolva o problemă. În plus, gândirea computațională se poate aplica și în alte domenii, nefiind strict legată de partea de tehnologie și programare.

Dezvoltarea gândirii computaționale

Rezolvarea problemelor prin logică

Programarea presupune dezvoltarea gândirii logice. Atunci când scriu cod, copiii învață cum să rezolve o problemă respectând niște pași. De asemenea, învață că, în programare, precum și în alte domenii, trebuie să formulezi ipoteze și să găsești soluții pentru fiecare parte componentă a problemei în parte.

Anticiparea problemelor prin gândire structurală

Prin programare, copiii învață cum să rezolve

erorile și cum să anticipeze eventualele greșeli. Ei învață cum să scrie cod în așa fel încât să nu apară erori în programul final. Programul este construit ca o casă, bucată cu bucată. Astfel, copiii învață cum să gândească structural, cum să așeze fiecare linie de cod pentru a forma un întreg, având în minte rezultatul final. Mai mult, abilitatea de a găsi soluții nu se aplică doar în domeniul IT, copilul dezvoltându-și capacitatea de a găsi soluții în general, indiferent de cariera pe care o va alege în viitor.

Dezvoltarea gândirii algoritmice

Algoritmii nu sunt specifici doar matematicii. Programarea se bazează, de asemenea, pe algoritmi. Gândirea algoritmică reprezintă abilitatea de a stabili pașii necesari rezolvării unei probleme, ajutându-i pe copii să rezolve problemele, la fel ca în matematică, pas cu pas, prin repetare, secvențiere și logică.

Dezvoltarea creativității

Copiii descoperă cum să-și folosească imaginația atunci când au la îndemână resurse limitate. În plus, dacă învățarea se face într-un mod distractiv, copiii sunt motivați să construiască ceva singuri. Atunci când vine vorba de programarea unor jocuri, copiii au posibilitatea să își pună creativitatea la bătaie. În plus, sentimentul că ceva făcut de ei funcționează, este poate cel mai bun mod de a crește stima de sine.



Pregătire pentru viitor

Este cert faptul că domeniul IT este un domeniu foarte bine plătit, care nu dă semne că ar putea stagna. La nivel mondial există deja o cerere de aproape un milion de programatori, ceea ce face din această meserie o carieră de viitor. Mai mult decât atât, pentru că a învăța programare dezvoltă multe abilități, tinerii au o perspectivă mai mare și un orizont mai larg atunci când vine vorba despre alegerea unei cariere, chiar dacă nu este în domeniul tehnologic.

Dezvoltarea gândirii diferite

Atunci când învață programare, copiii învață să nu gândească limitat. Practic, în acest domeniu, din nimic poți construi ceva. Deși trebuie respectați niște pași și niște norme, copiii pot să vină cu soluții diverse și chiar ieșite din comun pentru o problemă. Acest tip de gândire, în afara ”granițelor”, se poate apoi aplica în viața de zi cu zi. Copiii care învață programare sunt mult mai deschiși la nou și la găsirea de soluții inovative.

Indiferent dacă vor urma o carieră în tehnologie sau nu, copiii care învață programare își dezvoltă foarte multe abilități. Cert este că perioada cea mai bună pentru ca un copil să înceapă să învețe programare este vârsta de 6-7 ani, spun experții în domeniu.

Tendențe la nivel mondial în abordarea dezvoltării gândirii algoritmice

Dezvoltarea extrem de rapidă a domeniului IT a condus la creșterea numărului de locuri de muncă în acest domeniu. Marile companii (Microsoft, Apple, Google, Cisco etc.) au ajuns la concluzia că studierea disciplinei Informatică exclusiv în învățământul superior este tardivă pentru dezvoltarea competențelor legate de structurarea gândirii prin algoritmi și programare.

La nivel mondial s-a constituit o organizație formată din reprezentanți ai marilor companii din domeniul IT și din domeniul educațional (ministere ale educației, universități, licee, școli de stat și private) care susține introducerea Informaticii în curriculumul obligatoriu al școlilor începând încă din ciclul primar.

În viziunea aceste mișcări de proporții mondiale programarea este considerată o activitate strict necesară formării unei gândiri strategice, înțelegerii mecanismelor de bază ale lumii calculatoarelor.

Pentru promovarea introducerii algoritmicii ca disciplină de studiu la toate nivelele de studiu au fost organizate o serie de evenimente la care participă profesori și elevi din aproape toate țările lumii.

Săptămâna UE a programării – Code Week <https://codeweek.eu/>



Săptămâna UE a programării a fost lansată în 2013 de Young Advisors for the Digital Agenda Europe (Tineri consilieri pentru Agenda digitală pentru Europa).

Comisia Europeană sprijină Săptămâna codului, ca parte integrantă a strategiei sale pentru o “piață digitală unică”. Planul de acțiune pentru educația digitală al UE încurajează toate școlile din uniune să participe la eveniment.

Hour of Code

Ora de Programare reprezintă o introducere de o ora în știința computerului și programare, cu scopul de a demistifica limbajul specific și de a arăta că oricine poate învăța bazele acestuia. Încercarea de a apropia elevii de tehnologia computerelor și programare a devenit un efort global. Acesta începe printr-o parcurgere a unor activități de o oră, însă se extinde la tot felul de eforturi ale comunității.



Utilizarea aplicațiilor informatice la ciclul primar și gimnaziu

Scratch

Cu Scratch, îți poți programa propriile povești, jocuri și animații interactive și să distribuie creațiile tale către ceilalți în comunitatea online.



Scratch îi ajută pe cei mici să gândească creativ și sistematic, să lucreze colaborativ - lucruri

esențiale în secolul 21.

Scratch este un proiect al Lifelong Kindergarten Group de la MIT Media Lab. Este oferit gratuit.

Pe măsură ce creează povești interactive, animații, jocuri, muzică și artă cu Scratch, tinerii învață matematică și idei de calcul, concepte computaționale, cum ar fi iterația și condiționale. De asemenea, înțeleg concepte matematice importante cum ar fi coordonate, variabile și numere aleatorii.

Când elevii învață despre variabile în contextul Scratch, pot folosi imediat variabilele în diverse contexte: controlează viteza într-o animație, sau urmărește scorul în jocul creat.

De obicei, un student începe cu o idee, creează un prototip de lucru îl experimentează el, îl depanează atunci când lucrurile nu merg cum trebuie, primesc feedback de la alții, apoi o revizuiesc și o reprojetează. E o spirală continuă: obțin o idee, creează un proiect, ceea ce duce la idei noi, care duc la noi proiecte, și așa mai departe. Acest proces de proiectare combină multe dintre competențele de învățare din secolul 21 care vor fi

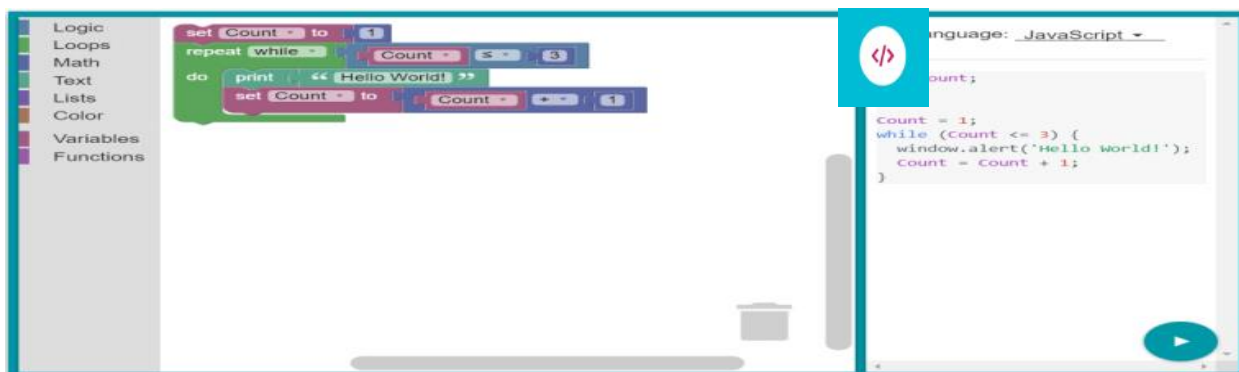
critice pentru succes în viitor: gândire creativă, comunicare, analizare sistematică, colaborare eficientă, proiectare în mod iterativ, învățare continuă.

Crearea de proiecte în Scratch ajută de asemenea elevii la atingerea unui nivel avansat de utilizare a tehnologiei digitale, de interacționare cu calculatorul și de asemenea îi ajută la atingerea unei fluente într-o limbă străină engleză, spaniolă sau altă limbă, o fluentă în exprimare în limba respectivă. Desigur, majoritatea elevilor nu vor ajunge programatori profesioniști, așa cum majoritatea nu vor deveni scriitori profesioniști, dar învățarea programelor oferă beneficii pentru toată lumea: permite elevilor să se exprime

ei înșiși mai mult și mai creativ, îi ajută să-și dezvolte gândirea logică și îi ajută să înțeleagă funcționarea noilor tehnologii pe care le întâlnesc peste tot în viața lor de zi cu zi.

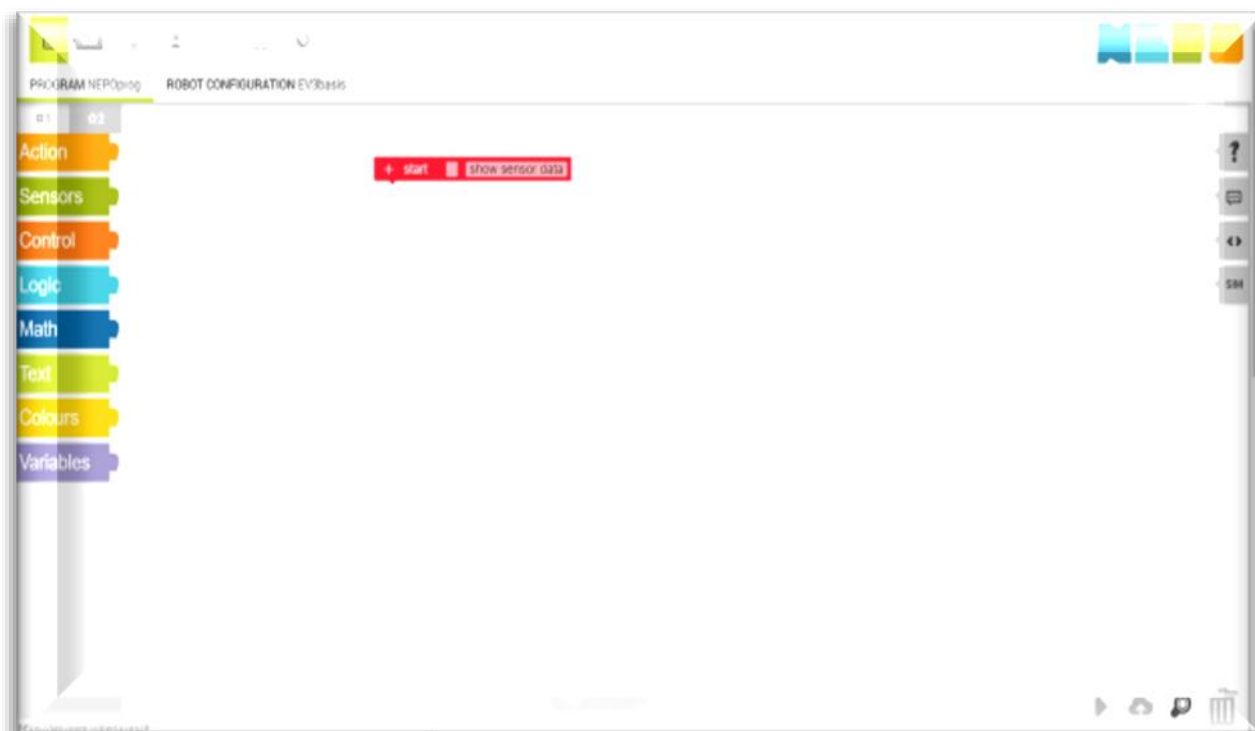
Blockly <https://developers.google.com/blockly/>

Blockly este un mediu vizual de dezvoltare a programelor, conceput de Google. Pentru scrierea codului programului sunt utilizate blocuri predefinite, grupate pe categorii de acțiuni. Diferența dintre Blockly și Scratch constă în faptul că Blockly oferă posibilitatea de a programa în mai multe limbaje de programare (JavaScript, Python, PHP etc.).



Blockly oferă copiilor biblioteci cu funcții predefinite din care se pot alege blocuri pentru realizarea programelor. Aplicația rulează în browser nefiind necesară instalarea ei.

Open Roberta Lab



Open Roberta Lab este un mediu de programare online ce permite programarea roboților virtuali 2D și reali 3D. Elevii pot realiza proiecte de mici dimensiuni în scopul familiarizării cu instrumentele puse la dispoziție de către mediul de programare.

Aplicația poate lucra cu roboți de tip Lego, (EV3 și WEDO), Arduino (Arduino, Raspberry Pi) etc. Platforma utilizează ca limbaj de programare limbajul NEPO. Punctul forte al acestui mediu de programare este acela că simulatorul integrat poate programa atât roboți reali, pentru care trebuie să existe un kit achiziționat, cât și roboți virtuali, în cazul în care elevul/școala nu deține un kit.

Acest fapt face ca predarea roboticii educaționale să fie posibilă în toate școlile din România prin utilizarea resurselor actuale ale școlilor, fără costuri suplimentare, folosindu-se doar calculatoarele din dotarea laboratoarelor de informatică și conexiunea la Internet.

Ar putea face parte disciplina Informatică din aria curriculară Matematică și științe?

Noțiunea de “algoritm” stă la baza conceptelor matematicii încă din antichitate, este utilizat în matematica actuală și este pus în valoare de către disciplina INFORMATICĂ, prin implementarea acestuia în programarea pe calculator.

În toate țările civilizate Informatica este o știință (Computer Science) - România o situează în aria curriculară Tehnologii.



PLATFORME EDUCAȚIONALE FOLOSITE ÎN PREDAREA ONLINE

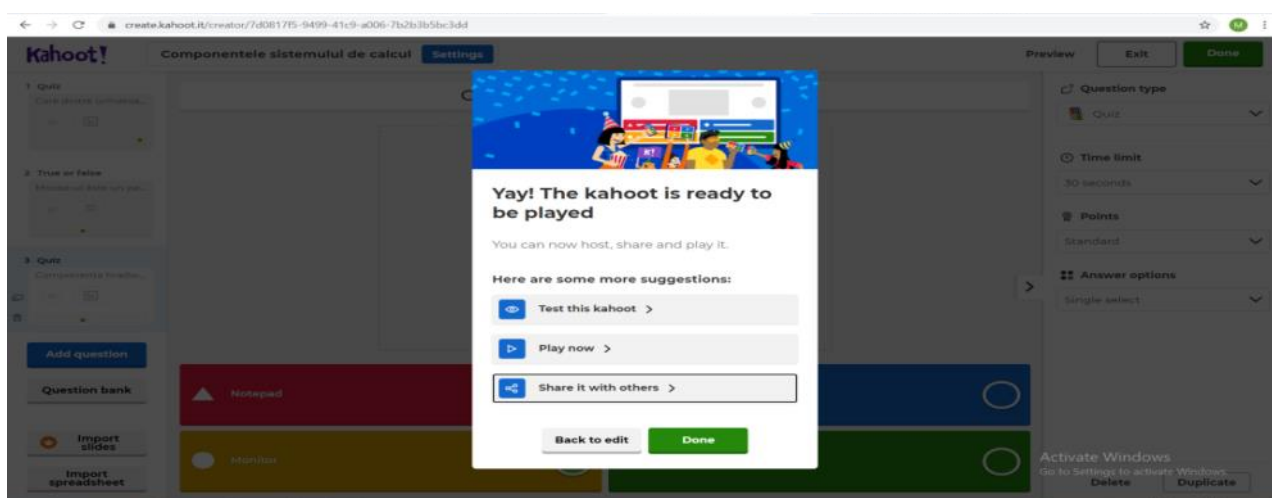
profesor Diaconu Mioara

KAHOOT este o platformă gratuită de învățare bazată pe joc și tehnologie educațională. Jocurile de învățare create pe această platformă pot fi create de oricine, pe orice subiect și prezintă avantajul că pot fi accesate folosind orice device(desktop, laptop, tabletă,telefon).

Pentru a utiliza această platformă, accesați linkul <http://getkahoot.com/> unde vom crea un cont personal ca profesor, care se poate și personaliza. Putem crea trei tipuri de Kahoot:

Quiz: test

Survey/Studiu



Discuție: sondaj de întrebări unice.



Dacă se dorește crearea unui Quiz, avem butonul CREATE: mai întâi acestui test i se va da un nume și apoi vom apela la tipurile de itemi puse la dispoziție de platformă și anume:

- * cu alegere duală(adevărat/fals)
- * cu alegere multiplă(cu un singur răspuns corect sau mai multe răspunsuri corecte)
- * tip puzzle
- * discuții
- * nor de cuvinte
- * întrebări deschise
- * brainstorm.

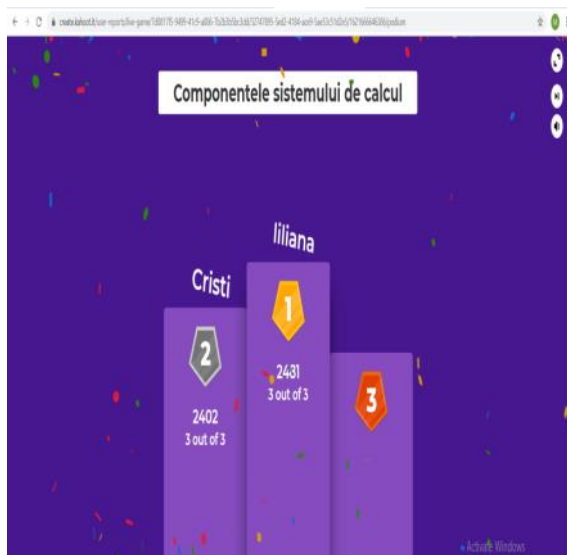
După introducerea tuturor întrebărilor se folosește butonul Done/salvarea care va determina apariția unei ferestre în care are se găsesc mai multe opțiuni și anume:

- * să testăm kahoot-ul/testul
- * să-l jucăm
- * să-l trimitem altora:elevilor, pe classroom, facebook, pe diferite grupuri.



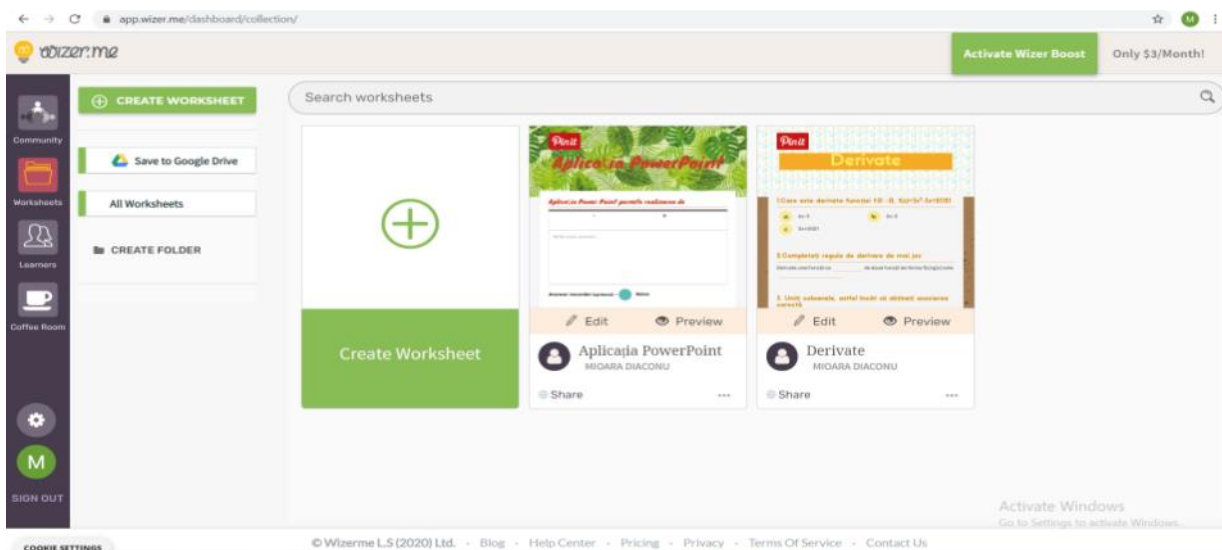
Alegem modul de testare calscic și apoi trimitem codul jocului. Jucătorii răspund întrebărilor de pe propriile dispozitive, în timp ce întrebările și răspunsurile sunt afișate pe un ecran comun, pentru a uni lecția. Jocul creaza un "moment-foc de tabără" , încurajând jucătorii să interacționeze. Rezultatele finale pot fi vizualizate de profesor prin intermediul unui raport, iar jucătorilor li se oferă chiar și un podium.

Importanța utilizării aplicației kahoot:



Obiectivitatea evaluării rezultatelor
Feed-back imediat pentru elev și profesor
Utilizarea acestor aplicații/hahoot-uri în evaluare încurajează crearea unui climat de învățare plăcut, relaxat.
Centralizarea și stocarea rezultatelor
Diversificarea modalităților de evaluare
Folosirea acestor aplicații îmbogățesc practica evaluativă, evitând rutina și monotonia
Interactiv.

WIZER.ME este o platformă unde profesorii construiesc fișe de lucru online, interactive și antrenante. Fișa de lucru permite inserarea de imagini, linkuri, videoclipuri. Poate lua forma unei ore cu toți pașii de predare-învățare-evaluare. Elevii își pot crea conturi și pot rezolva fișa direct pe platformă. Astfel, în contul cadrului didactic se adaugă fiecare material lucrat de către elevi. Acesta poate să intervină, adăugând comentarii, corectând, adăugând feedback. Sarcinile pot fi interactive, diversificate, pot include numeroase exerciții-joc, antrenante, captivante pentru elevi. Poate fi o variantă foarte utilă pentru încărcarea unei lecții, respectând toate etapele acesteia.

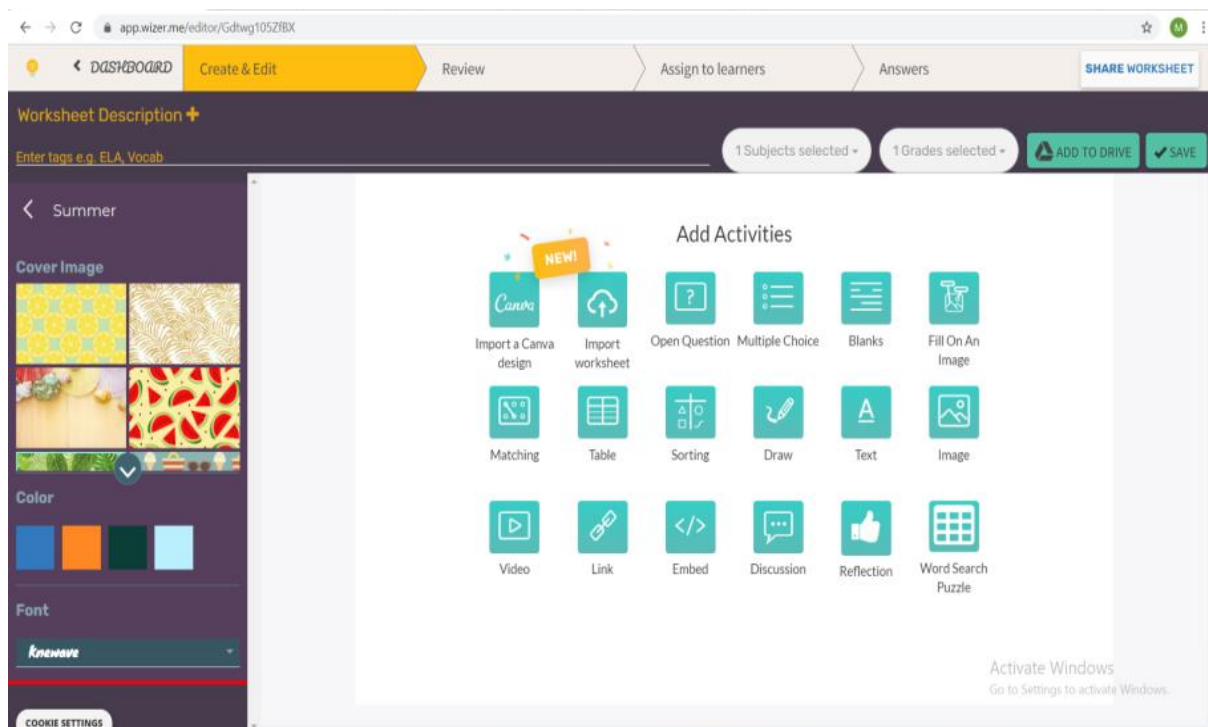


Pentru realizarea unui material util la clasă pentru diverse momente ale lecției ne vom face un cont pe această platforma <https://app.wizer.me/> ca profesor. Platforma pune la dispoziție fișe de lucru lucrate de alții pe care le putem folosi sau adapta utilizând butonul *Community*, dar putem crea fișe de lucru personalizate utilizând butonul *Worksheets*. Cu ajutorul instrumentului *Create worksheet* putem crea o fișă de lucru nouă pe care o putem personaliza în funcție de conținutul informației prezentate.

Activitățile care pot fi folosite în fișile de lucru sunt:

- * Întrebare deschisă
- * Întrebări multiple
- * text în care elevii vor completa spațiile libere (itemi de completare)
- * completarea unor texte pe o imagine
- * asocierea unor noțiuni (itemi de asociere)
- * tabele în care elevii vor completa noțiuni studiate după anumite criterii
- * inserarea unor obiecte ca imagini, texte, link-uri, videoclipuri
- * selectare sau trierea dintr-o listă
- * cuvinte într-un puzzle

Salvarea fișei de lucru se realizează cu ajutorul butonului *Save*, previzualizarea acesteia de la eticheta *Review* iar transmiterea elevilor cu ajutorul etichetei *Assign to learners* către clase din classroom sau cu ajutorul unui link (<https://app.wizer.me/learn/TUUBSS>) care se trimite pe grupul elevilor. La eticheta *Answers* putem vizualiza răspunsurile, putem trimite feedback elevilor chiar și tipări răspunsurile.



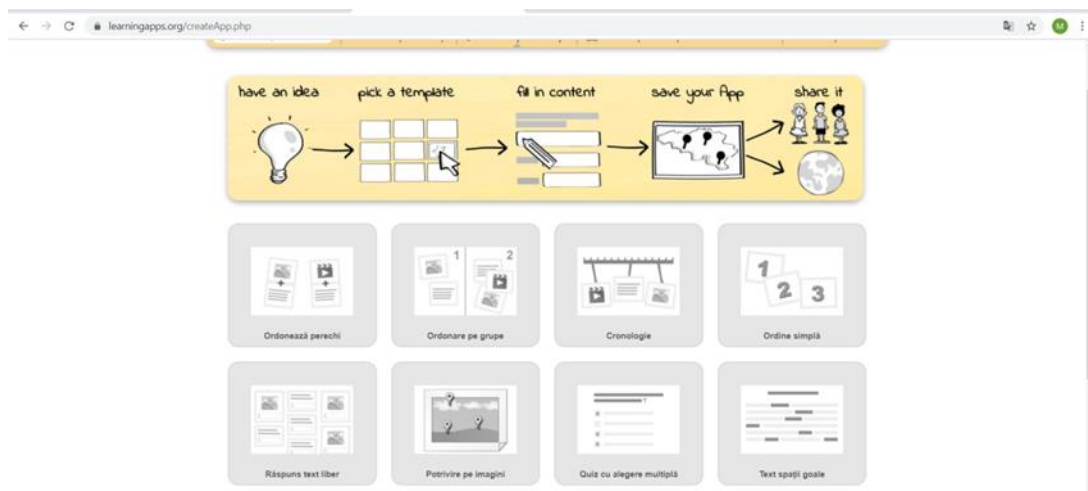
Elevii își pot crea conturi și pot rezolva fișa direct pe platformă. Astfel, în contul cadrului didactic se adaugă fiecare material lucrat de către elevi. Acesta poate să intervină, adăugând comentarii, corectând, adăugând feedback.

LEARNING.APPS este platformă gratuită, care permite crearea de module de învățare – fiecare unealtă este considerat un bloc care poate fi integrat alături de alte blocuri. Un alt avantaj al platformei este faptul că permite inserarea de fișiere audio-video care se găsesc deja pe internet, permite un management al clasei mai eficient – de la înregistrarea elevilor care poate fi făcută de către profesor, până la statistica activităților desfășurate în această clasă. Learningapps este o platformă folosită pentru crearea de mici joculețe care pot fi folosite în consolidarea cunoștințelor prin exerciții. Este foarte atractivă pentru elevi întrucât aceștia pot executa sarcinile atât singuri, cât și în competiție cu alți colegi. La final pot vedea rezultatul.

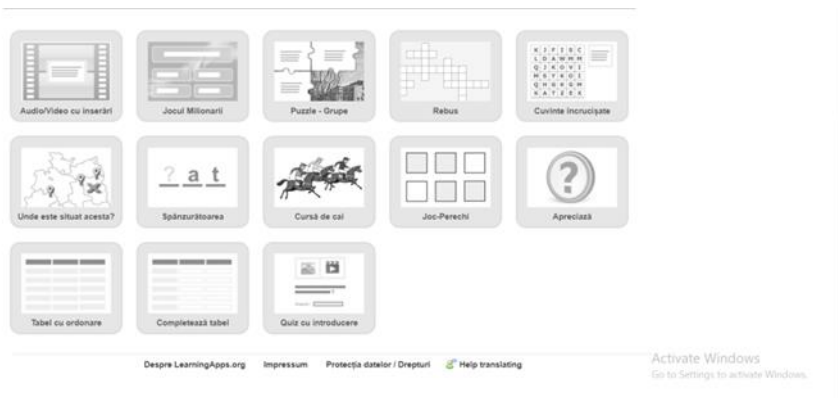
Pentru a putea folosi această platformă pe care o găsiți la adresa www.learningapps.org va trebui să ne facem cont, în care avem la dispoziție o bază de date a materialelor realizate de alți utilizatori.



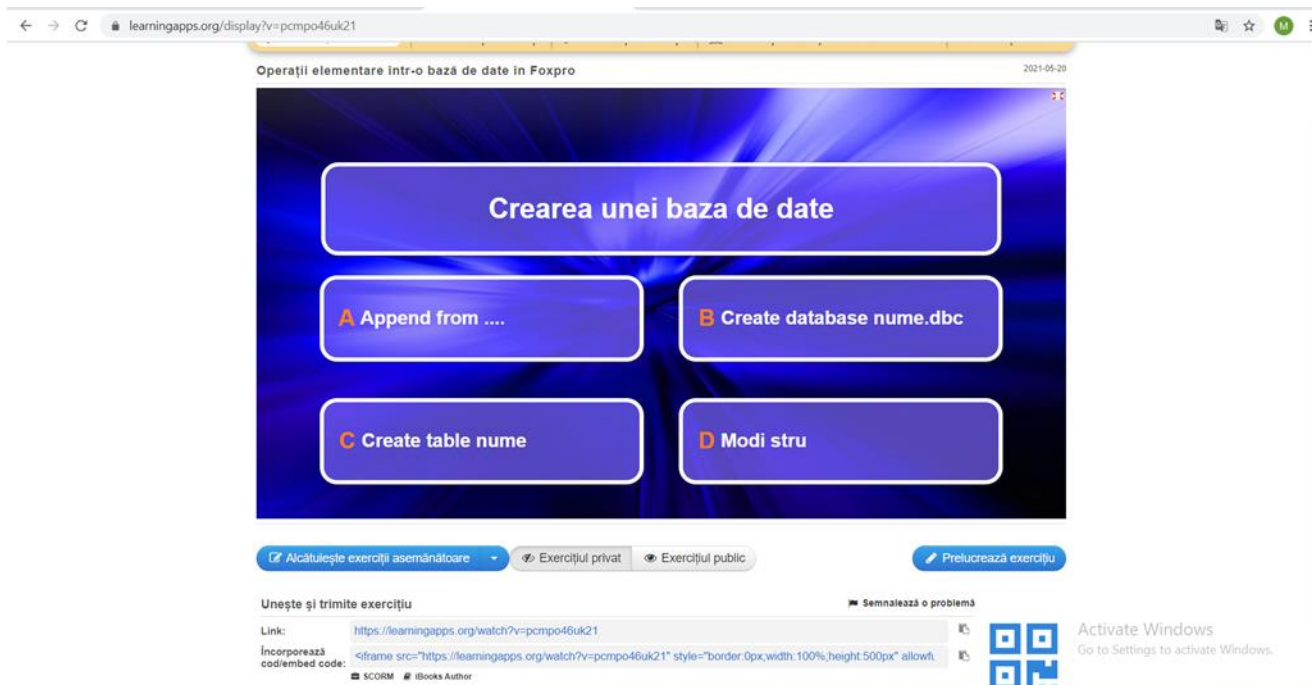
Pentru crearea de exerciții noi vom accesa opțiunea Alcătuieste exercițiu după care vom alege un template sau o formă de joc după cum se va vedea în figurile următoare:



Tipurile de exerciții care pot fi alcătuite sunt diverse pornind de la: Cuvinte încrucișate; Jocul Milionarii; Marchează în text; Quiz cu alegere; Quiz cu alegere; Completează tabel; Quiz cu introducere; Grilă ordonare; Joc-Perechi(Perechi imagini); Joc-Perechi(Text/Imagine); Ordonare cu hartă; Ordonare grupe; Ordonare pe imagini; Ordonează perechi; Puzzle – Grupe; Tabel ordonare; Rebus; Spânzurătoarea; Text spații goale; Completează tabel; Quiz cu introducere; Audio/Video cu inserări; Avizier; Caiet; Calendar; Chat; Mindmap; Scrieți împreună.



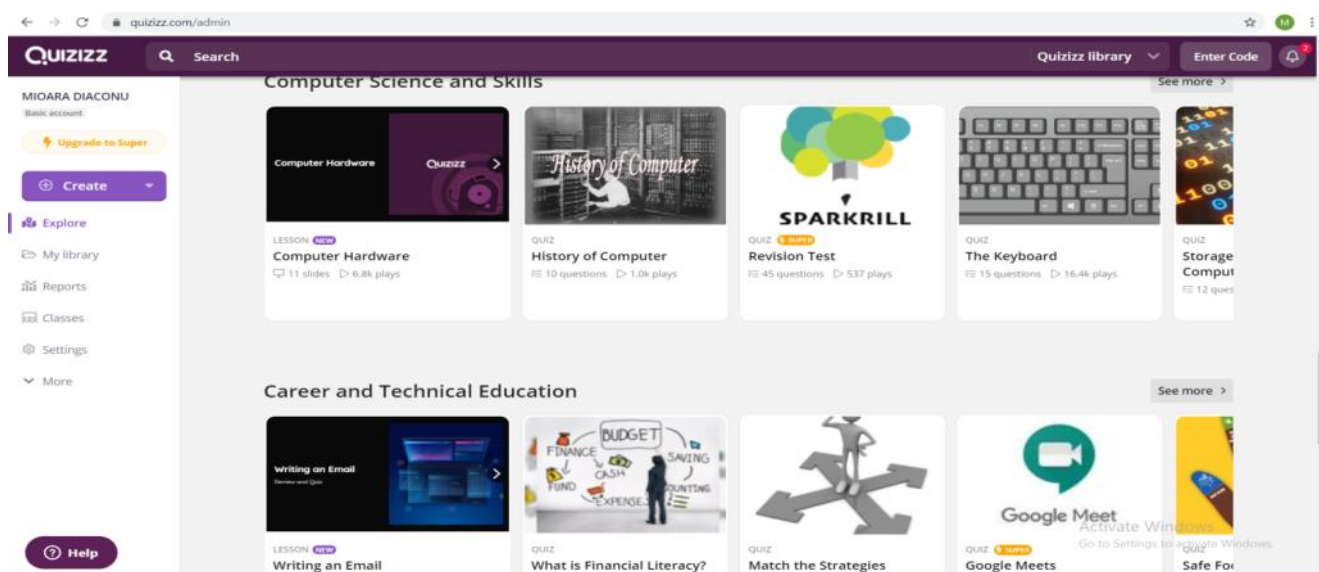
În fereastra nouă vom introduce numele exercițiului nou și vom introduce pe rând elementele de care avem nevoie în cadrul lecției și le previzualizăm, corectăm dacă este cazul și salvăm exercițiul. În parte de jos apare link-ul (<https://learningapps.org/watch?v=pcmpo46uk21>) pe care îl vom trimite elevilor spre rezolvare(acesta de mai jos este *Jocul milionarilor*).



Cu ajutorul statisticilor, le putem urmări munca și rezolvarea problemelor.

LearningApps este un ajutor pentru profesori și pentru elevi. Profesorul are posibilitatea să structureze materialul pe care dorește să-l predea în cel mai bun mod, sau să se folosească de exercițiile oferite de colegi. Pentru elevi, LearningApps oferă moduri cât mai interactive și atrăgătoare de însușire a noilor informații. Prin intermediul LearningApps elevul va învăța mult mai ușor prin descoperire și mai ales prin interactivitate. Avantajul major al acestei aplicații este acela că se distribuie gratuit și cu codul sursă deschis pentru utilizator. Acest fapt permite utilizarea cu mult mai eficientă. Astfel utilizatorul poate schimba modulele existente ale aplicației sau poate adăuga module noi având posibilitatea de a extinde funcționalitatea(<https://learningapps.org/watch?v=pa26kcx3321>).Aplicația LearningApps este simplă pentru folosire, se integrează ușor în procesul de învățămînt și se adaptează foarte ușor pentru orice programă a oricărei instituții de învățămînt.

Quizizz.com este o platformă gratuită de evaluare a claselor care permite tuturor elevilor să învețe împreună. Platforma conține o bibliotecă cu lecții și teste grupate pe diferite domenii și discipline, create de utilizatori, pe care le putem utiliza modifica/adapta după necesitatea lecției la clasă. Testele create cu această aplicație pot fi accesate de către elevi de pe telefoanele mobile, tablete, laptop sau calculatoare conectate la Internet. Pentru a utiliza această platformă vom face un cont pe <https://quizizz.com> și în urma logării fereastra conține materiale create de ceilalți utilizatori, precum și rapoarte, clase, iar în My library sunt materialele create de mine.

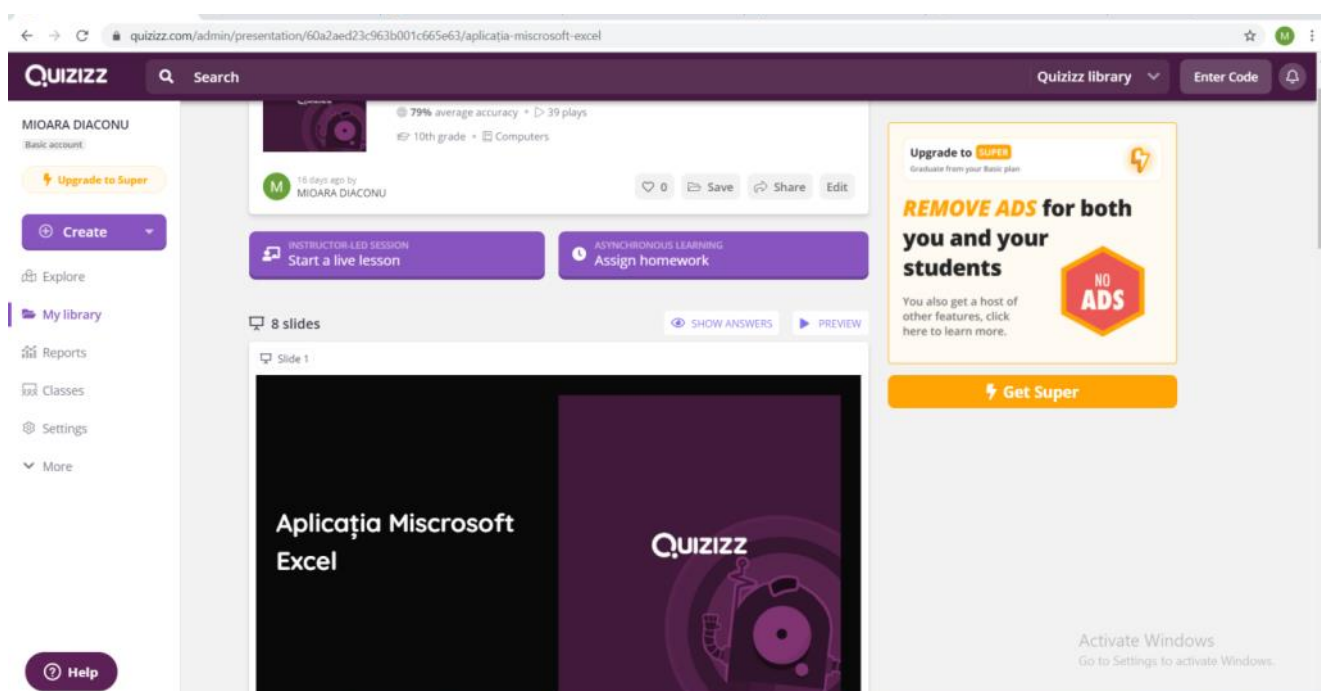


Crearea unui nou material vom utiliza butonul *Create* care ne permite să alegem test sau lecție. La opțiunea test platforma ne permite utilizarea următorilor itemi:

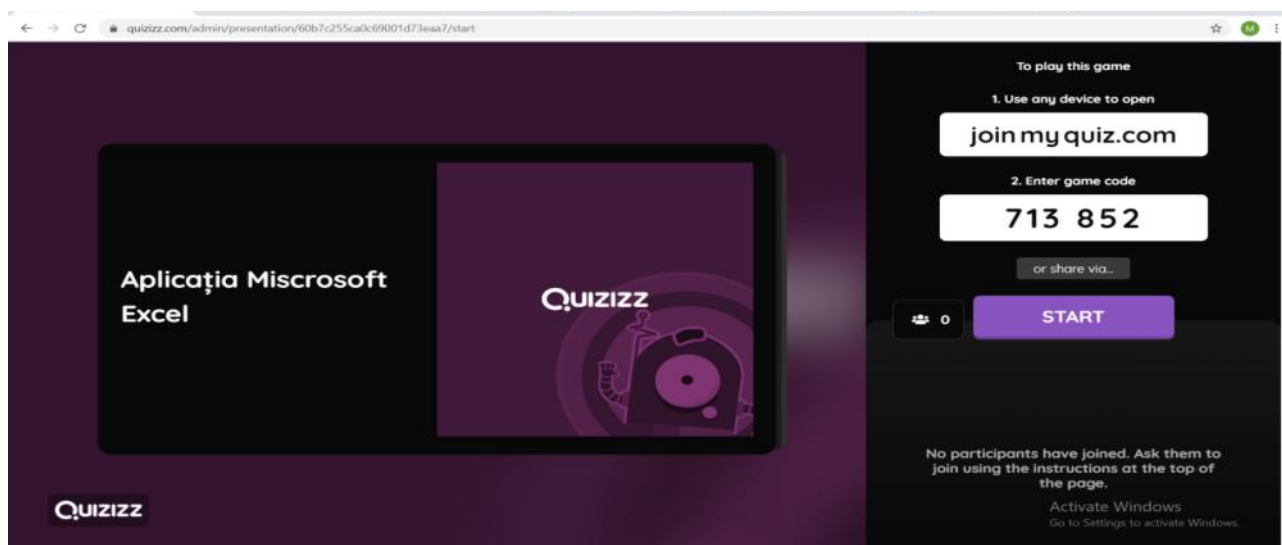
- ◇ itemi cu alegere multiplă cu un singur răspuns corect(Multiple choice)
- ◇ itemi cu mai multe răspunsuri corecte(Checkbox)
- ◇ itemi de completare(fill in the blank)
- ◇ poll(dicuții)
- ◇ întrebare deschisă(Open-ended)



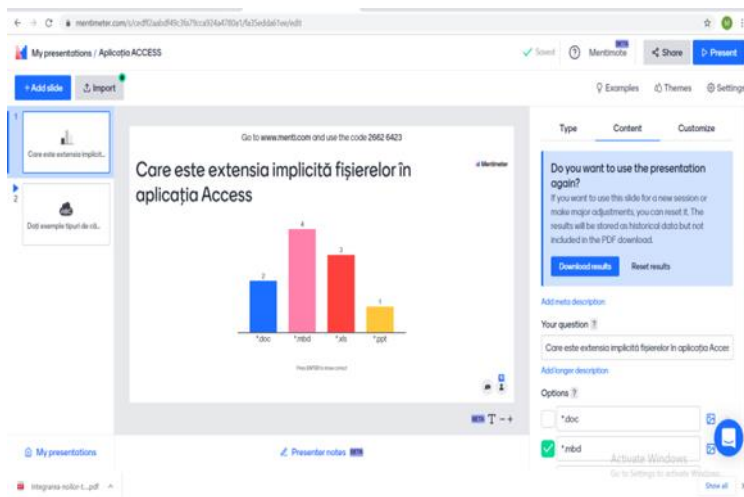
În casetele de text se completează conținutul întrebării *Write your question here* apoi se adaugă, pe rând, posibilele răspunsuri. Răspunsurile pot fi sub formă de text sau sub formă de imagini. Se bifează răspunsul corect și se setează timpul acordat rezolvării apoi se salvează *Save*. Numărul de întrebări pe care trebuie să le aibă un test este minim 6. Se adaugă, pe rând, toate întrebările testului și se apasă butonul *FINISH QUIZ*. În continuare se adaugă informații suplimentare testului: se setează nivelul (clasa) pentru care a fost creat testul, se setează domeniul și categoria la care poate fi asociat testul creat apoi se salvează *Save quiz info*. Testul poate fi trimis unui profesor sau mai multor profesori *Share* (nu este obligatoriu!) După finalizarea acestor pași testul poate fi rezolvat de elevi: *Live Game* (în timp real, în clasă, cu toți elevii), *Homework* (temă), *Solo Game* (fiecare independent de ceilalți).



La opțiunea *Start a live lesson* fereastra este cea de mai jos în care putem trimite elevilor link-ul și codul elevilor sau *or share via* către clasele elevilor sau pe grupul acestora respectiv butonul *START* care determină pornirea testului.



Pe măsură ce elevii se conectează cu codul primit acestia vor apărea în lista testului. Când toți elevii sunt conectați se apasă butonul **START** și elevii primesc testul pentru a-l rezolva. Rezultatele se pot vedea în timp ce elevii rezolvă în secțiunea **Reports**. La finalizarea testului se apasă butonul **END GAME**. Datele conținute în secțiunea **Reports** pot fi printate direct din aplicație **Print**, pot fi descărcate **Download** într-un fișier **Excel**, pot fi trimise prin email tuturor părinților **Email all parents**.



Pentru a utiliza platforma vom folosi link-ul <https://www.mentimeter.com/> pentru profesori unde ne vom face un cont iar pentru elevi [menti.com](https://www.menti.com).

Pentru a crea o nouă prezentare vom utiliza butonul **New presentation**, apoi se va alege numele acesteia. În partea dreaptă va fi meniul pentru editarea tipului de întrebare. După această alegere veți pune titlul întrebării și opțiunile de răspuns și modul în care vor fi afișate. Ultimul

pas este prezentarea, ce se realizează cu ajutorul butonului "Present" din dreapta sus. După prezentarea întrebării în partea de sus a ecranului va fi vizibil un cod. Elevii vor accesa pagina aplicației și vor introduce codul afișat în spațiul destinat acestuia, această acțiune fiind precedată de răspunderea întrebării pe dispozitivul propriu și afișarea rezultatului pe ecranul cu întrebarea.

La eticheta **type** platforma pune la dispoziție următoarele tipuri de diapozitive:

- ◇ alegere multiplă
- ◇ nor de cuvinte
- ◇ întrebare deschisă
- ◇ clasament
- ◇ întrebări și răspunsuri
- ◇ selectare de răspunsuri
- ◇ imagini
- ◇ videoclipuri



La eticheta **Content** se introduce conținutul diapozitivului și fac setările corespunzătoare, salvarea realizându-se automat. Trimiterea acestei prezentări către clasă sau grupul de elevi se realizează cu ajutorul butonului **Share** prin care se realizează o copie link-ului. Rezultatele sunt trimise în format excel cu ajutorul opțiunii **Export results**.

Este o sarcină importantă pentru noi profesorii să motivăm, să inițiem și să menținem motivația. Motivația în didactică este un principiu și sunt foarte multe metode. Educația asistată de software are un efect motivant din toate privințele.

Mult succes la utilizarea aplicațiilor prezentate mai sus, care vă va ajuta să creați situații educative care vă pot fi de ajutor la lecție, predare, învățare, exersare sau aplicare!



5

dintre cele mai bune programe de completare Google Slides, pentru a crea o prezentare impresionantă

**Prof. informatică Mardare Liliana,
Liceul Tehnologic „Elena Cragiani”, Tecuci**

Pear Deck – for Google Slides

Acest instrument este excelent pentru cei care doresc să aibă o interacțiune în prezentarea lor Google Slides. Pear Deck adaugă obiecte interactive prezentării, cum ar fi alegerea multiplă, introducerea textului, introducerea numerică, desenul și activitățile glisabile. Pear Deck vine cu o varietate de opțiuni pentru efectuarea cursurii la clasă, inclusiv integrarea cu Google Apps, opțiunea de a importa PowerPoint și PDF, prezentări sau documente, posibilitatea de a adăuga videoclipuri YouTube la sesiunile de clasă și mai mult. Puteți realiza, de asemenea întrebări cu variante multiple și de a obține un răspuns instantaneu de la elevii dvs. printr-un sondaj on-line.

Pentru a începe, fie se deschide un **PowerPoint** sau **PDF** existent, sau, puteți crea propriile slide-uri din interiorul Pear Deck. (fig.1). Suplimentul conține o bibliotecă (Our Template Library) cu slide-uri predefinite pentru diferite momente ale lecției (Beginning of Lesson, During Lesson, End of Lesson), pentru dezvoltarea învățării (Learning Development – Critical Thinking, Social - Emotional Learning), pentru diferite discipline (matematică, științe, limbi străine, etc).

Puteți crea oricând propria activitate personalizată sau întrebări de la zero, utilizând itemi de tip text, cu răspuns de tip număr, alegere multiplă, website, drawing și draggable pentru a implica elevii și pentru ca activitatea didactică să fie cât mai atractivă și interactivă. Cele două opțiuni drawing și draggable sunt disponibile numai făcând upgrade la Pear Deck Premim.

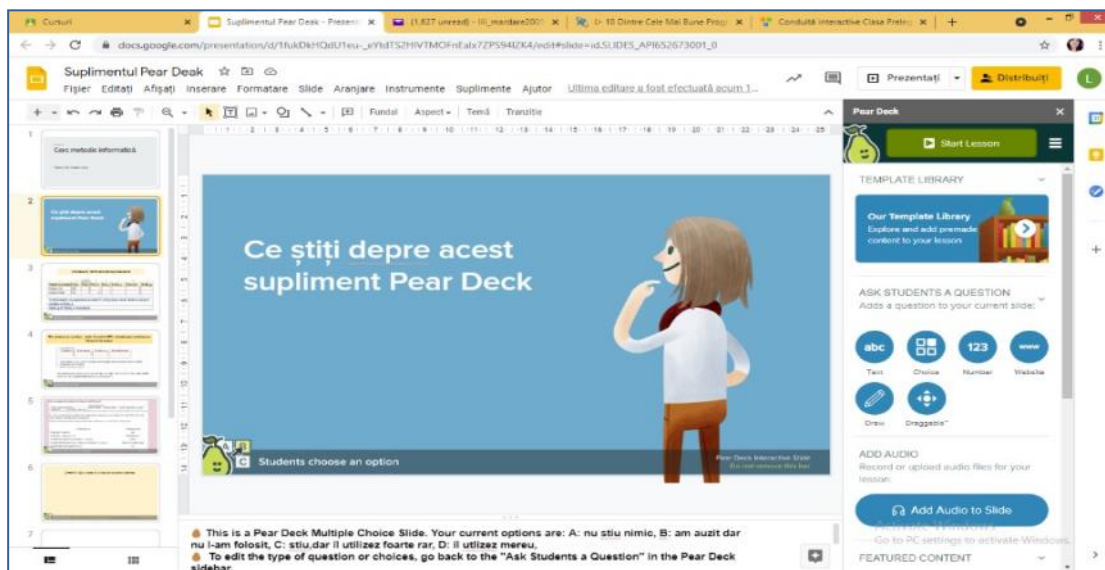


Fig.1

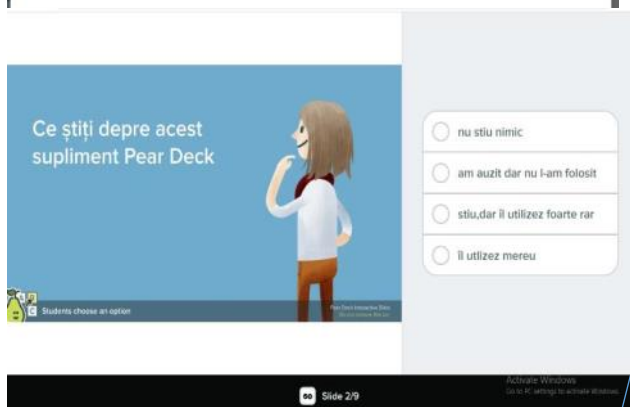
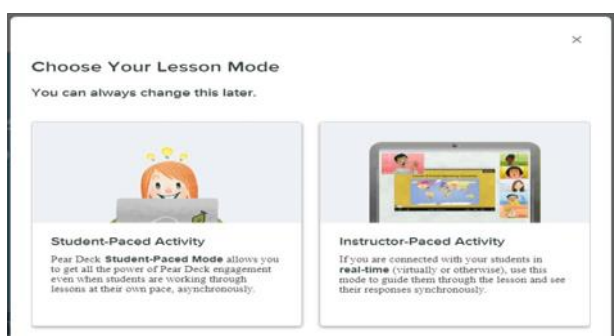


AUDIO LA ORICE TIP DE SLIDE

Cu ajutorul programului Pear Deck, puteți lăsa mesaje audio, descrieri sau exemple pe care elevii dvs. să le audă în timp ce citesc instrucțiunile și răspund la întrebări interactive din prezentările dvs. Pear Deck Slides. Puteți realiza o nouă înregistrare instantaneu sau puteți încărcați una de pe computer, acestea vor fi redată în vizualizarea elevilor, deci este perfectă pentru orice lecție de învățare la distanță!

START LESSON

Începeți prezentarea cu click pe butonul **Start Lesson** (activitatea poate fi în modul asincron sau în timp real) și elevii se vor alătura (Give students link) sesiunii dvs. de pe orice dispozitiv cu un browser web și vor răspunde la întrebările afișate (vizibile în modul de vizualizare pentru elevi). Răspunsurile lor pot fi afișate în mod anonim pe proiectorul clasei, sau pe ecranul computer-ului, dacă dorim ca elevii doar să lucreze împreună și să-și exprime ideile, sau ei pot apărea cu numele și prenumele lor (în acest caz este necesar să se logheze cu user-ul de pe platforma google classroom), atunci când dorim să facem o evaluare (curentă, formativă, sumativă).

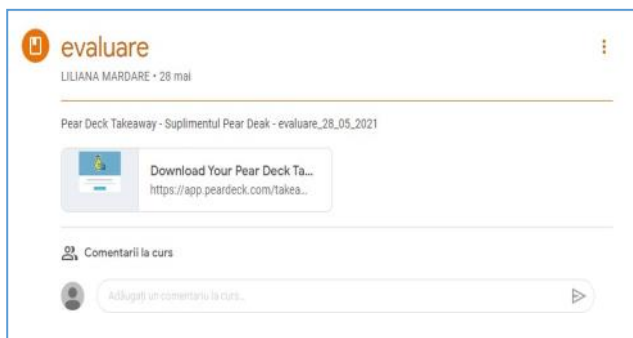


Modul de vizualizare pentru elevi

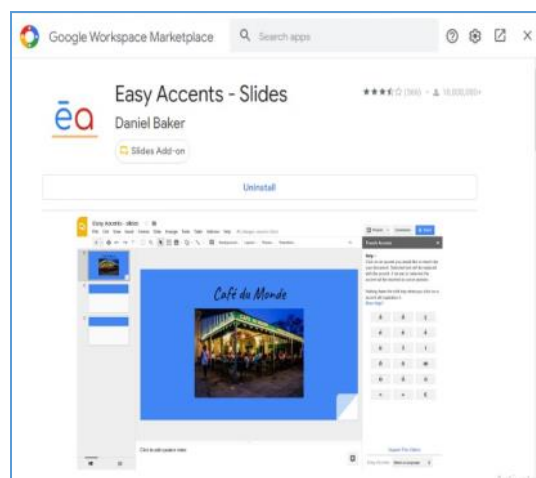
Modul de vizualizare pentru profesori



END OF LESSON



La sfârșitul lecției toată activitatea va fi salvată și poate fi postată pe google classroom, și în contul drive. Profesorul va putea analiza răspunsurile elevilor și le va putea evalua.



Easy Accent Slides

Este extrem de util dacă creați adesea

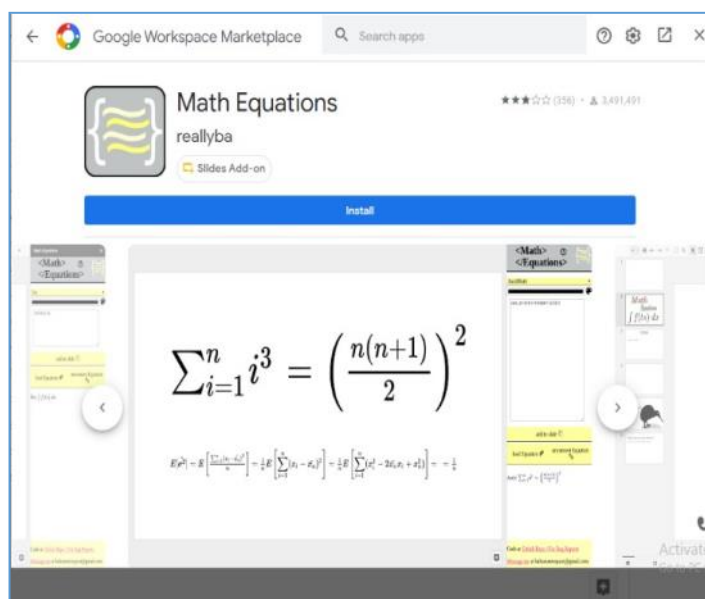
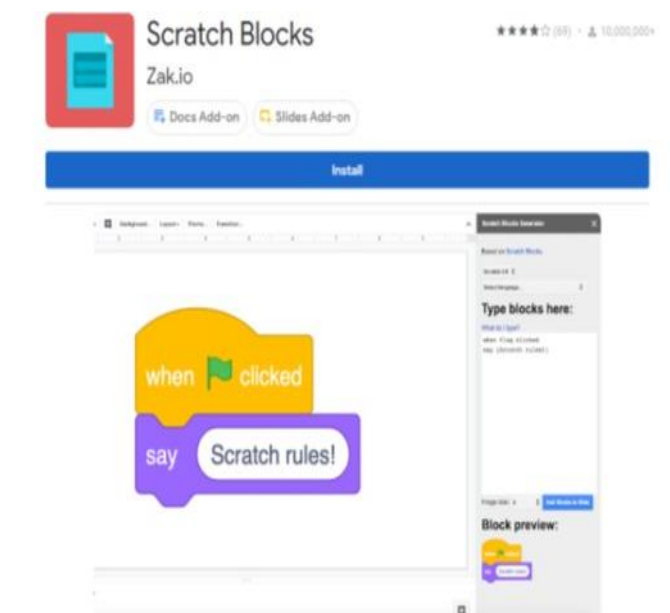


Photo slideshow - Vă permite să importați în bloc imagini din Google Drive în prezentarea dvs., câte o imagine pe fiecare diapozitiv. Este o modalitate simplă de a crea o prezentare de diapozitive, în loc să adăugați fotografiile pe rând.



Scratch Box – puteți utiliza instrumentele Scratch direct în Google Slides.

Instrumentele sunt suplimente terță parte create pentru a îmbunătăți performanța și funcționalitatea atunci când utilizați Prezentările Google. Ele vă fac prezentările mai impresionante, interactive și atractive. Pentru a le obține, tot ce aveți nevoie este să accesați “Descărcați suplimente” și să răsfoiți colecția, sau să o căutați pe cea dorită. Instalați programul de completare și acceptați permisiunile și sunteți gata de acțiune!



Introducerea limbajului Python la gimnaziu a fost o gură de oxigen pentru programarea din România, fiind o decizie predictibilă și înțeleaptă. Haideți să analizăm aceste afirmații: la ora actuală Python este un limbaj de programare flexibil, ușor de învățat și larg folosit în industrie. Dacă ne uităm la facultățile de profil, putem constata o orientare către acest limbaj, nu numai în introducere în programare, dar și în studierea structurilor de date și mai ales a algoritmilor fundamentali. De exemplu, Facultatea de Matematică și Informatică din cadrul Universității București dar și unele secții de master ale Facultății de Informatică din Iași, se axează pe acest limbaj de programare, la fel universitățile din Cluj, Timișoara sau Brașov.

Instituții de renume mondial precum Michigan State University (CSE, 2014), Massachusetts Institute of Technology (Guttag, 2013), New York University, Carnegie Mellon University și University of Cambridge. Și liceele ar trebui să reacționeze la această tendință de predare

în Python? Python este un limbaj de programare adecvat pentru toți elevii de liceu?

Cum am descoperit Python? Provoocat de elevii, pentru că un profesor bun este inspirat de elevi și devine pe zi ce trece tot mai competitiv, fără a stinge pasiunea pentru cunoaștere a elevilor, am început să predau concepte de bază în Python. În plus, există numeroase tutoriale și platforme care oferă resurse Python. Dintre acestea voi discuta despre câteva:

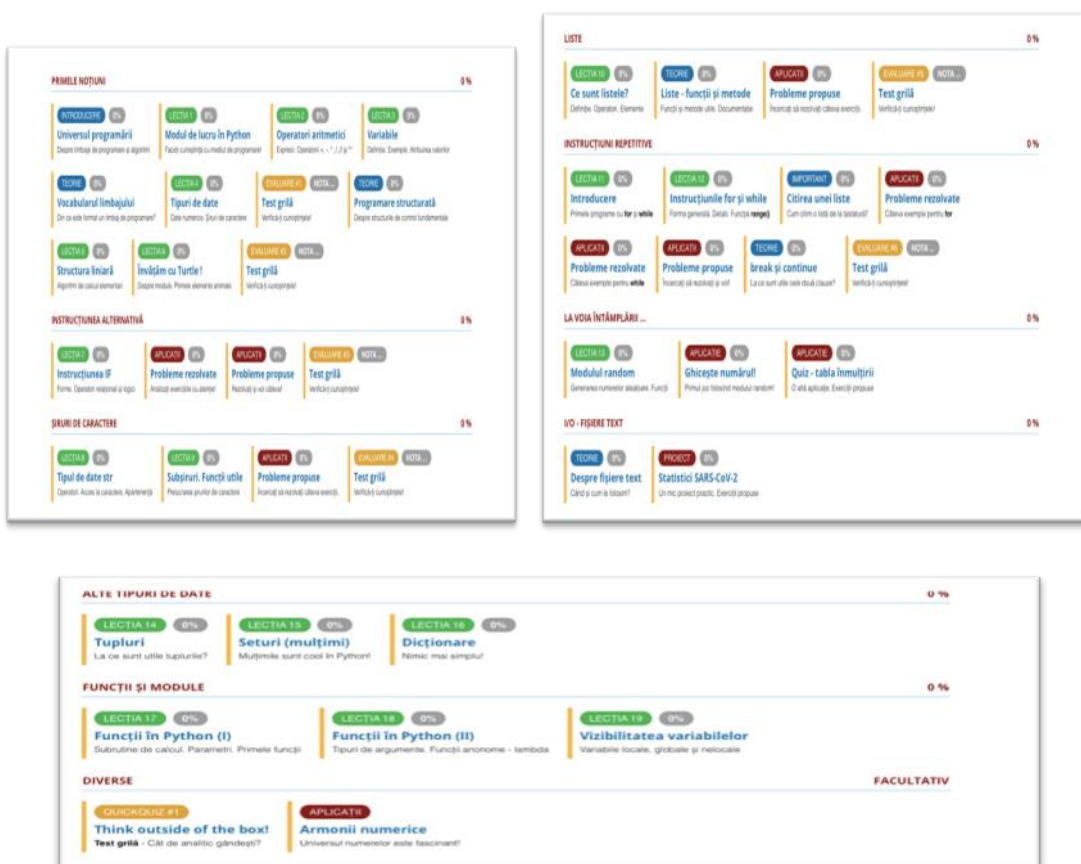
Youtube- Este fără îndoială cea mai largă și mai complexă resursă pentru a preda și învăța un limbaj de programare. Evident că nu înlocuiește un profesor de informatică, dar elevii din ziua de astăzi învață foarte ușor și din surse nebănuite. Aici sunt numeroase canale, un canal valoros este cel oferit de www.freecodecamp.org, dar există evident și alți programatori sau canale dedicate. Aici întâlnim cursuri de câteva ore sau mai complexe, cel mai important după părerea mea este să exersăm și să realizăm un portofoliu pe o



O altă resursă valoroasă atât pentru elevi cât și pentru profesori este situl realizat de prof. dr. Vlad Tudor www.pythonisty.ro care prezintă în mod riguros și plăcut modalități de exersare a elementelor de limbaj în Python precum și o miniculegere de probleme.



Aici există și un club iar în viitor se vor putea realiza și clase asociate unui profesor. Printre subiectele dezbătute:



Alte resurse gratuite pentru studierea Python :

- * <https://www.w3schools.com/python/default.asp>
- * www.coursera.org
- * www.edx.org
- * Kanaacademy

Bineînțeles, există și cărți despre Python, o primă resursă recomandată unui începător este cartea „*Learn Python in one day and learn it well*” scrisă de James Chan și care abordează într-o manieră destul de simplă de parcurs, în limba engleză, etapele învățării acestui limbaj de programare. Să facem o scurtă trecere în revistă a acestui limbaj.

Pentru a programa în Python avem nevoie de un interpretator, cel mai la îndemână fiind disponibil direct de pe site-ul oficial www.python.org. De aici descărcăm ultima versiune Python, care la momentul scrierii acestui articol era 3.9.5. Putem utiliza consola pentru a executa comenzi secvențial sau putem realiza scripturi, adică fișiere salvate cu extensia .py. Prima comandă studiată este cea de afișare în consolă, după care sunt descrise modalități concrete de realizare a scripturilor Python. Spre deosebire de C++, acest limbaj este interpretat, ceea ce îi conferă o oarecare rapiditate. Personal prefer Pycharm pentru a executa programele scrise în Python, care are o versiune gratuită pentru elevi (Community edition). Visual Studio Code oferă suport pentru Python și reprezintă de asemenea o alternativă pentru IDLE.

Cartea prezintă noțiuni de bază despre variabile, modul de definire și inițializare al acestora, dar și operatorii din Python. Spre deosebire de C++, în momentul definirii unei variabile, nu este necesar să stabilim tipul acesteia și mai mult decât atât, tipul unei variabile se poate modifica pe parcursul execuției unui program.

Un capitol aparte este dedicat tipurilor de date utilizate în Python, întregi, reale, șiruri, liste, tuple și dicționare. Ulterior, cartea continuă cu prezentarea funcțiilor `input()` și `print()`, a comentariilor pe o linie și mai multe linii de cod și urmează structura firească a studierii unui limbaj de programare: structuri de control, funcții și module, lucrul cu fișiere, prelucrarea fișierelor precum și proiecte care utilizează instrucțiunile prezentate anterior.

Cartea se termină printr-o serie de anexe care prezintă lucrul cu string-uri, liste, tuple și dicționare.

Ceea ce este interesant în această carte este maniera explicită în care sunt prezentate informațiile, îmbinate armonios cu exemple practice, care consolidează noțiunile de programare.

Acest articol nu se dorește o prezentare amănunțită a limbajului de programare Python ci doar o provocare spre studiu. Aceste lucruri fiind spuse, spor la programare în Python!

https://www.researchgate.net/publication/309622733_Is_Python_an_Appropriate_Programming_Language_for_Teaching_Programming_in_Secondary_Schools



SOCRATIVE – Instrument digital pentru realizarea evaluării sumative

Prof. Anghelache Iulia-Nela

SOCRATIVE este o aplicație pentru evaluare - o alternativă la testele clasice, pe suport de hârtie. Se recomandă utilizarea ei cu predilecție în lecțiile de recapitulare - pentru evaluarea cunoștințelor sau în lecțiile de predare-învățare-evaluare, pentru obținerea unui feed-back imediat (în partea de fixare a cunoștințelor). Este un instrument ușor de utilizat, cu ajutorul căruia se obțin rezultate în timp real sau rapoarte care pot fi descărcate ca fișiere Excel ori pdf. Cu **Socrative** se pot crea și administra teste pentru diferite discipline, care pot fi copiate sau partajate cu alți profesori.

Avantaje ale utilizării aplicației:

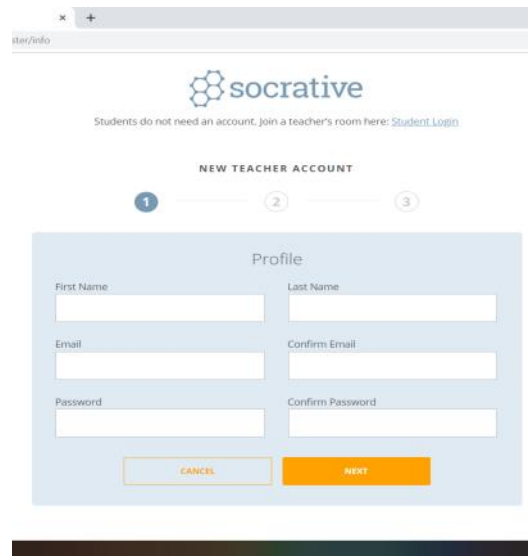
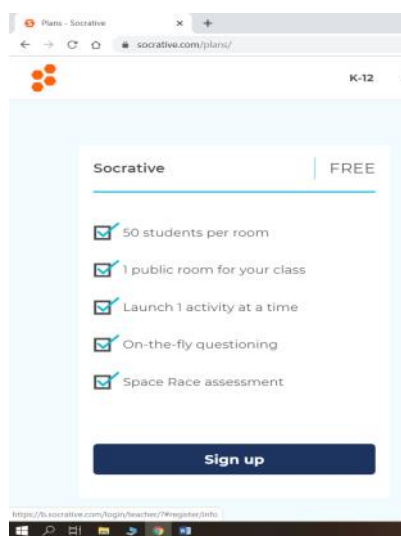
- * Permite crearea de teste cu 3 tipuri de itemi: variante multiple de răspuns, adevărat/fals, cu răspuns scurt.
- * Rezolvarea testului de către elevi se poate face utilizând diferite dispozitive IT (PC, laptop, tabletă, telefon mobil), cu condiția conectării la internet.
- * Elevii nu au nevoie de cont pentru a rezolva testul propus de profesor. Este suficient să acceseze aplicația **SOCRATIVE STUDENT** și să se conecteze la clasa profesorului (room), pe baza codului transmis de acesta.
- * Elevii pot rezolva testul oriunde (acasă, în sala de clasă, în parc, la bibliotecă)

Aplicația **Socrative** poate fi accesată pe site-ul www.socrative.com. Sunt 3 planuri (versiuni) ale aplicației, dintre care unul singur este gratuit. Acesta oferă unui utilizator cu cont **TEACHER** următoarele: o cameră publică pentru clasa sa, 50 de elevi pe sesiune, rapoarte, partajarea testelor prin codul SOC și altele. În continuare, voi prezenta pașii care trebuie urmați pentru accesarea versiunii Free a aplicației **Socrative**, în calitate de utilizator **TEACHER**.



Pasul 1. Crearea contului

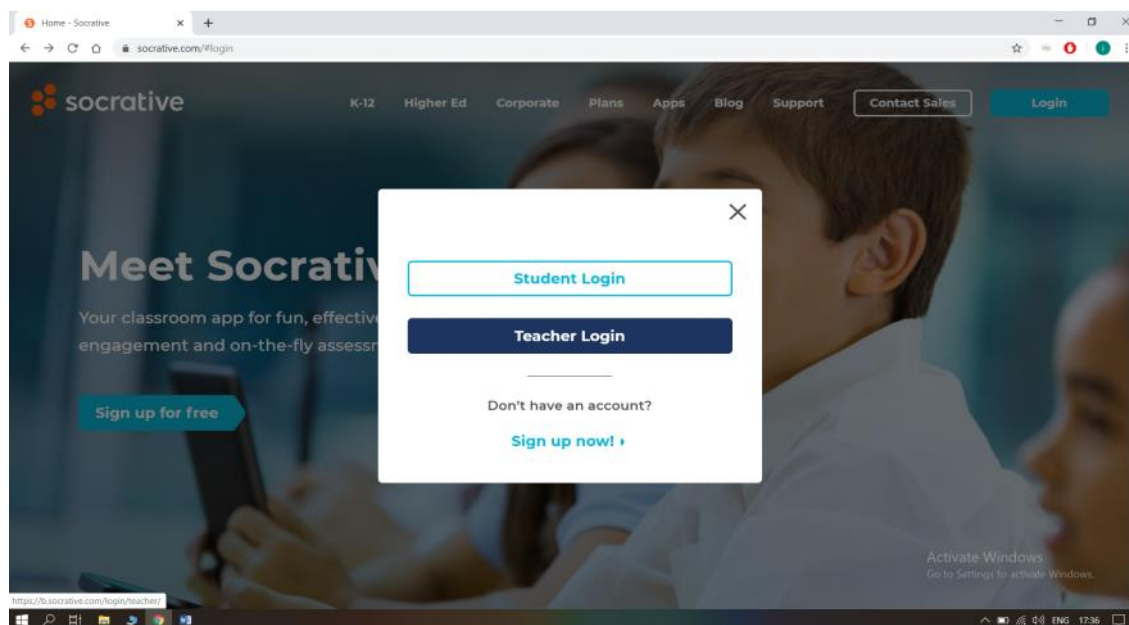
Se dă click pe butonul **SIGN UP FOR FREE**. Pe coloana corespunzătoare planului *Free* se dă click pe butonul **SIGN UP**. Se deschide pagina **NEW TEACHER ACCOUNT**:



Pasul 2. Se completează toate câmpurile și se apasă butonul **Next**, pentru a finaliza crearea contului.

Pasul 3. Conectarea la aplicație

Pentru conectarea ca profesor, se dă click pe butonul **TEACHER LOGIN**, situat în partea dreaptă sus. În caseta **Teacher Login** se introduc adresa de email și parola de la Pasul 2.

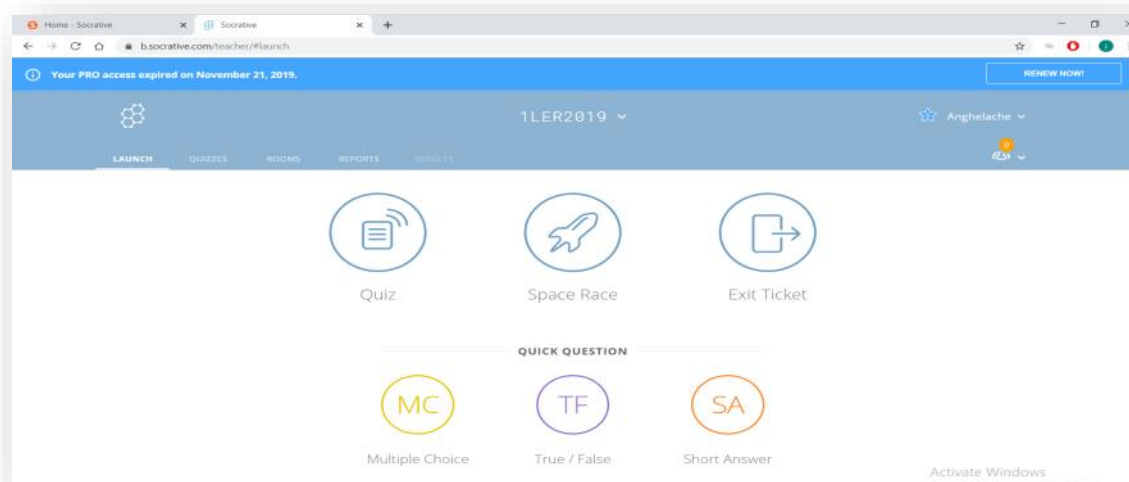


Pentru conectarea ca elev, **STUDENT LOGIN**, profesorul comunica elevilor codul camerei format numai din litere mari și cifre (de exemplu în cazul meu, **1LER2019**). Acest cod poate fi modificat din secțiunea **Room**.

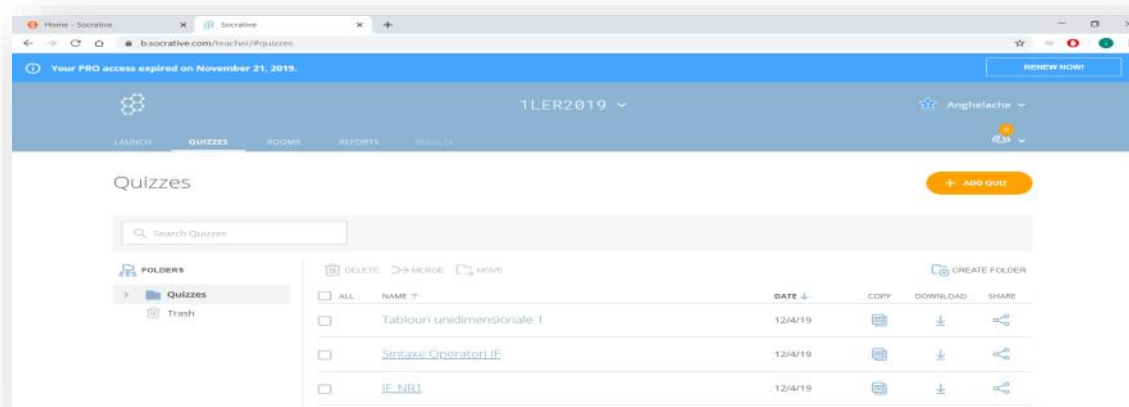


Interfața este împărțită în 5 secțiuni, ce se pot accesa dând click pe fiecare, în bara de sus:

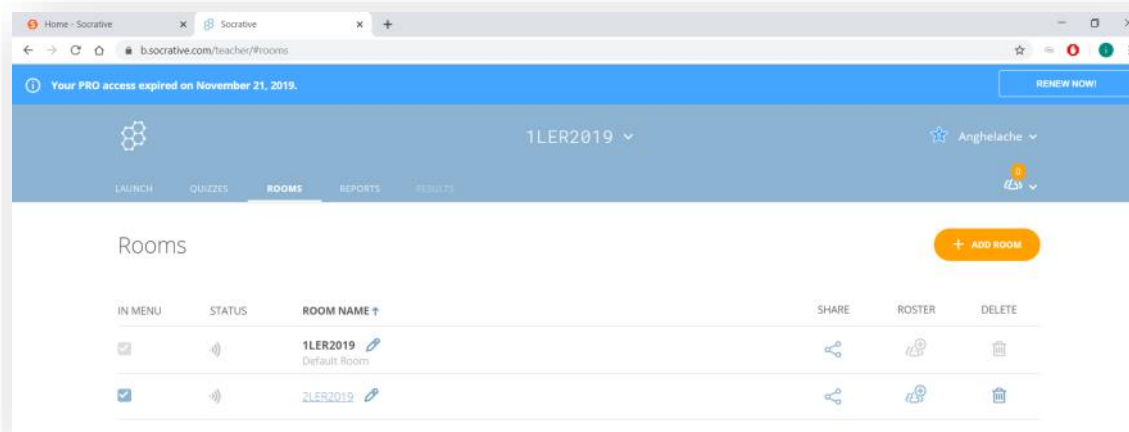
Launch – lansarea testelor, are opțiuni privind modalitatea de lansare și feedback.



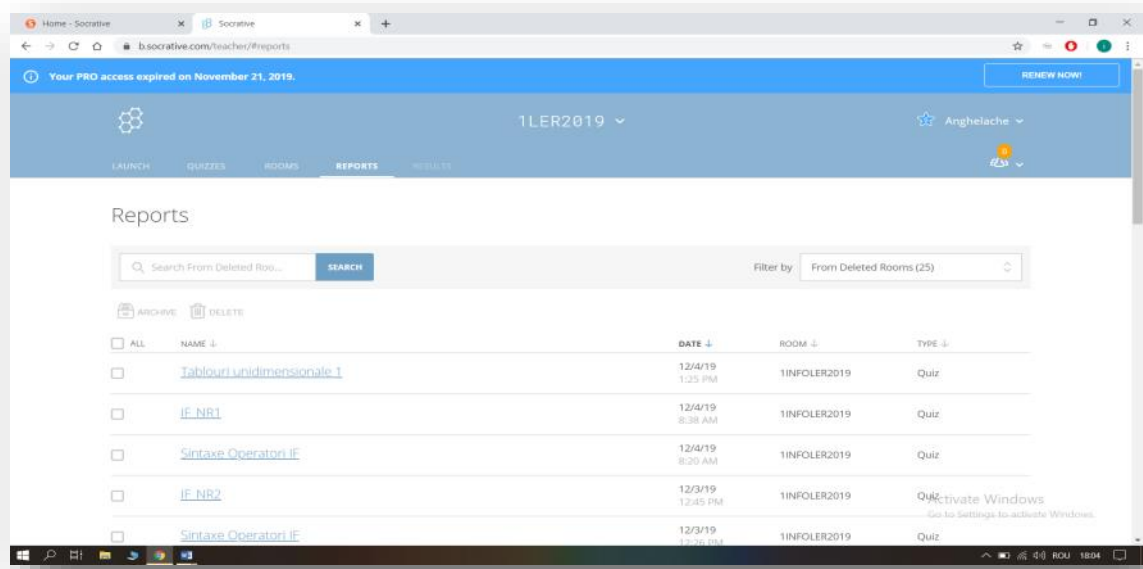
Quizzes – permite crearea testelor; are opțiuni de creare sau import a testelor, afișează toate testele create și importate.



Room – clasa creată (numele acestuia se poate redenumi):



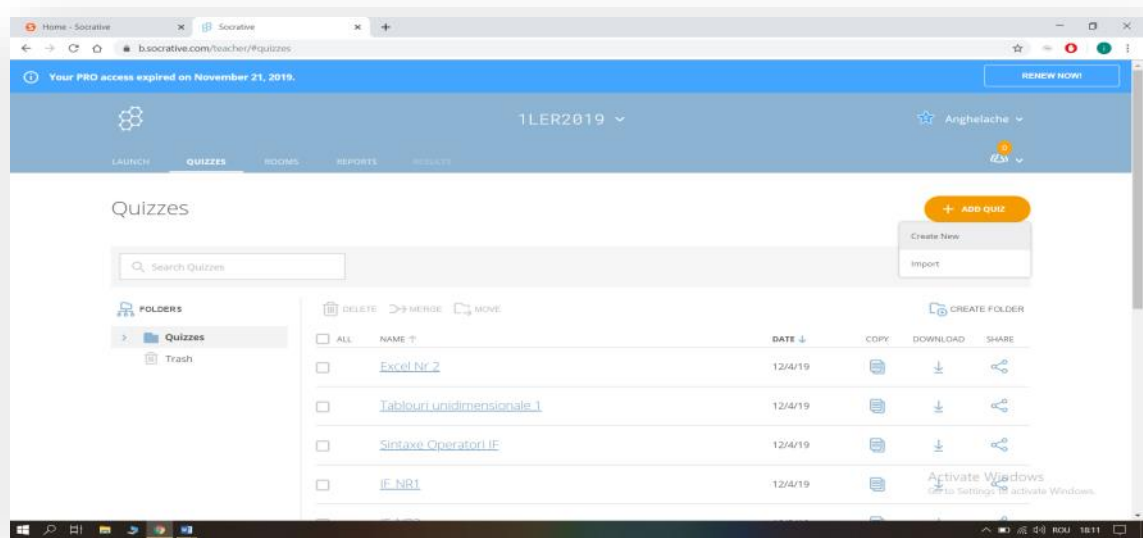
Reports prezintă rapoarte care pot fi descărcate, salvate în Google Drive sau trimise prin email



Results afișează rezultate în timp real, în timpul sesiunii active, după lansarea testului.

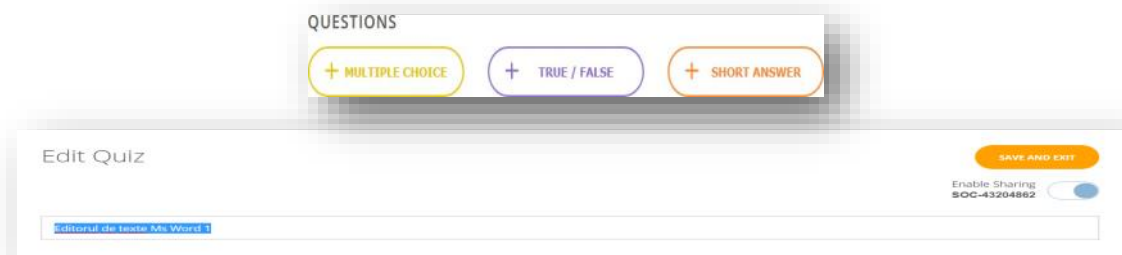
Crearea unui test

Pasul 1. În secțiunea **Quizzes** se apasă butonul **Add Quiz** și se alege opțiunea **Create New** pentru a crea un test nou. Cu opțiunea **Import**, se poate importa un test din **Socrative Shared Quiz List**, pe baza codului SOC al testului.



Pasul 2. Se deschide pagina de creare a testului. Fiecărui test creat i se atribuie în mod automat un număr SOC cu ajutorul căruia testul poate fi căutat în resurse. Mai întâi, trebuie introdus titlul testului.

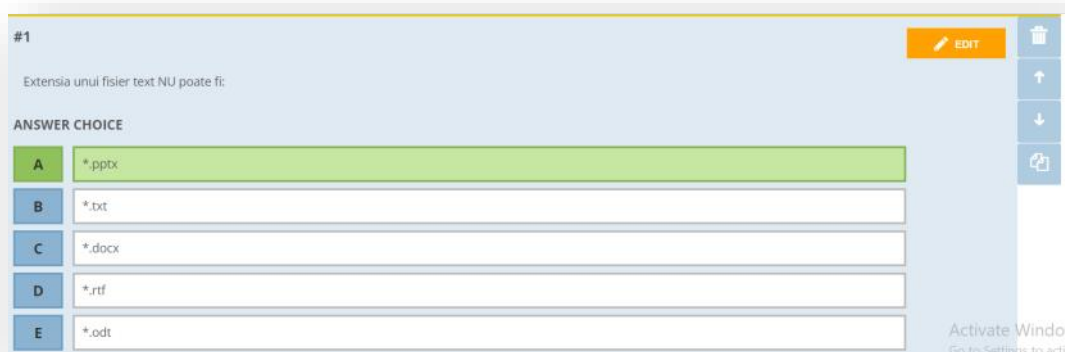




Sunt disponibile 3 tipuri de întrebări: cu alegere multiplă, adevărat/fals și cu răspuns scurt.

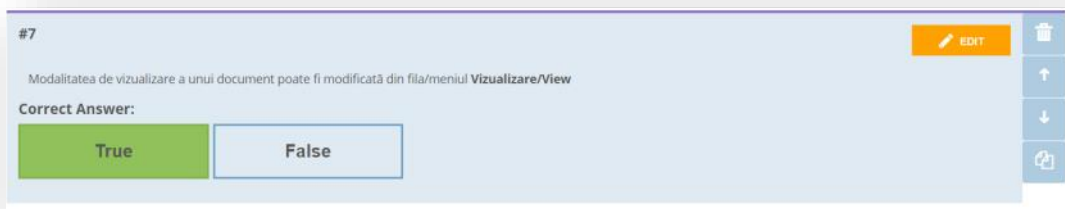
Pasul 3. Crearea întrebărilor cu alegere multiplă

Se dă click pe butonul + **MULTIPLE CHOICE**, apoi se completează întrebarea și variantele de răspuns, bifând varianta corectă. Se pot adăuga alte variante de răspuns cu ajutorul butonului **ADD ANSWER**. Se apasă **Save** pentru salvarea întrebării. Modificarea întrebării se poate face cu ajutorul opțiunii **EDIT**.



Pasul 4. Crearea întrebărilor de tip Adevărat / Fals

Se selectează + **TRUE / FALSE**, apoi se completează întrebarea, se bifează răspunsul corect (**True** sau **False**) și se apasă **Save**.

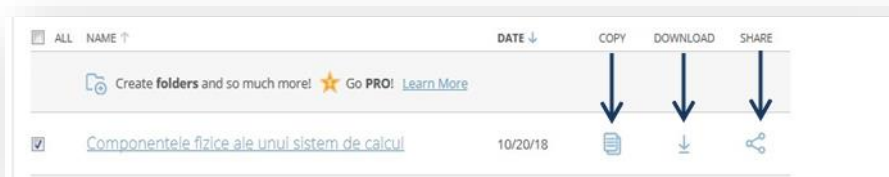


Pasul 5. Crearea întrebărilor cu răspuns scurt

Se selectează + **SHORT ANSWER** și se introduce întrebarea. Se poate insera o imagine cu ajutorul butonului din stânga casetei de text corespunzătoare întrebării:

Se repetă pașii 3, 4, 5 pentru a crea alte întrebări și se salvează fiecare întrebare. O întrebare salvată poate fi modificată cu ajutorul opțiunii **EDIT**. Ordinea întrebărilor din test poate fi schimbată cu ajutorul butoanelor săgeți (- și +). La final, se dă click pe **SAVE & EXIT**.

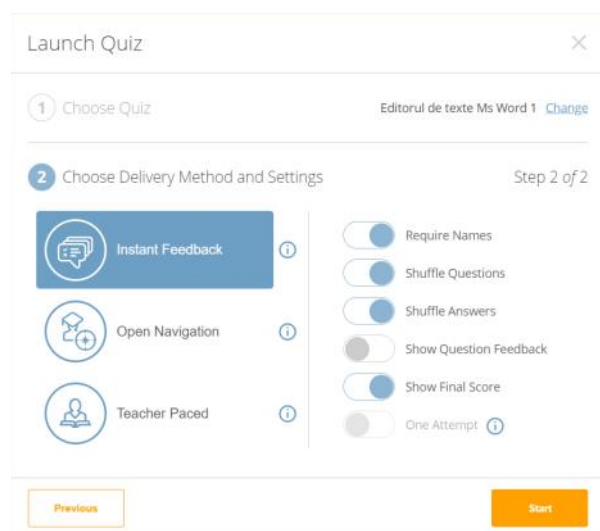
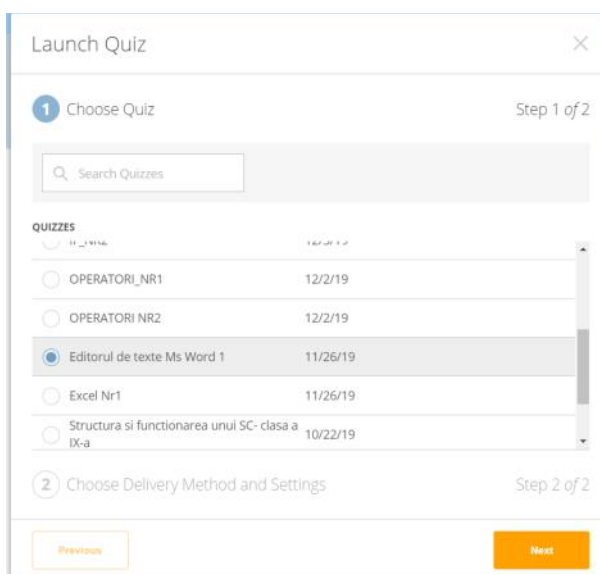
Un test creat poate fi copiat, partajat sau descărcat în format pdf pentru a fi tipărit. Pentru realizarea acestor operațiuni, în pagina **Quizzes** se selectează testul și se apasă pe pictograma corespunzătoare (**COPY**, **SHARE** sau **DOWNLOAD**) din partea dreaptă.



Lansarea testului

Pasul 1. În secțiunea **LAUNCH** se apasă pe **Quiz** și, în caseta **Choose Quiz**, se selectează testul, apoi se dă click pe **NEXT**.

Pasul 2. În caseta **Choose Delivery Method and Settings**, se alege una din cele 3 opțiuni: **Instant Feedback**, **Open Navigation** sau **Teacher Paced** și se pot selecta opțiunile din dreapta: **Require names**, **Shuffle Questions**, **Shuffle Answers**, **Show Question Feedback**, **Show Final Score**. Se apasă butonul **Start** pentru a lansa testul. Se va deschide secțiunea **Results**, în care vor fi afișate rezultatele elevilor care au susținut testul.



Editorul de texte Ms Word 1

FINISH

Show Names Show Answers

Name ↑	Progress (%)	1	2	3	4	5	6	7	8
Popa	0%								
Class Total									

Click question numbers or class total percentages for detailed views.



Inteligența Artificială

Prof. Zeld Jenica, Liceul Teoretic Dunărea

S-a vorbit prima dată de Inteligența Artificială (AI – Artificial Intelligence) în 1956, când totul părea o utopie, un vis prea frumos pentru a fi realizat, un stadiu al dezvoltării considerat a fi greu de atins. În ultimii aproape 50 de ani, termenul a prins contur, devenind realitate, fiind în prezent folosit în toate științele care doresc să se afirme. Inițiatorul său, **prof. John McCarthy** a prezentat noul concept în vara anului 1956 la întrunirea “Darthmouth Summer Research Project on Artificial Intelligence”.

Inteligența Artificială poate fi definită ca simularea inteligenței umane procesată de mașini, în special, de sisteme de calculatoare. Acest domeniu a fost, în general, caracterizat de cercetări complexe în laboratoare și doar destul de recent a devenit parte a tehnologiei în aplicații comerciale.

Principalele abilități inteligente **umane** pe care trebuie să le dovedească un sistem AI – un computer sau un robot coordonat de un computer – sunt:

abilitatea de a descoperi sensul într-o situație dată, abilitatea de a generaliza, plecând de la un caz particular, capacitatea de a învăța din experiențe anterioare.

Pe lângă cele de mai sus, iată alte câteva comportamente asociate cu inteligența umană: planificarea, rezolvarea problemelor, percepția, manipularea și – poate una dintre direcțiile cele mai evidente spre care se îndreaptă

Inteligența Artificială – inteligența socială și creativitatea.

DOMENIILE INTELIGENȚEI ARTIFICIALE

Inteligența artificială cuprinde:

- **Jocuri:** programarea calculatoarelor să joace jocuri cum ar fi șahul sau jocul de dame;
- **Sisteme expert:** programarea calculatoarelor să ia decizii în situații reale (din viața reală) – de exemplu anumite sisteme expert ajută doctorii să stabilească diagnosticul bolii pe baza simptomelor.

• **Limbaje naturale:** programarea calculatoarelor să înțeleagă limbajul uman natural.

• **Rețele neurale:** sisteme care simulează inteligența prin încercarea reproducerii tipurilor de conexiuni fizice întâlnite în creierul animalelor.

• **Sisteme fuzzy:** numite și sisteme cu mulțimi vagi bazate pe o logică în care între cele două valori de adevăr: adevărat și fals există o infinitate de valori ce pot modela mai bine lumea reală;

• **Algoritmi genetici:** algoritmi bazați pe modul de dezvoltare evoluționist copiind multe din principiile geneticii;

• **Robotică:** programarea calculatoarelor să vadă, să audă și să reacționeze alți stimuli senzoriali;

• **Noi tehnici ale domeniului inteligenței artificiale** care apar și se dezvoltă continuu indicând marele interes de care se bucura această știință.

La începutul anilor 1980 s-a crezut că sistemele expert vor constitui viitorul inteligenței artificiale și ale calculatoarelor în general. Până acum însă aceste sisteme n-au răspuns așteptărilor. Multe sisteme expert ajută experții umani în diferite domenii cum ar fi medicina sau ingineria dar aceste sisteme sunt foarte scumpe și sunt de folos doar în situații speciale. Astăzi cel mai fierbinte domeniu al inteligenței artificiale este cel al **rețelelor neurale** care sunt folosite cu succes în anumite domenii cum ar fi recunoașterea vocii și procesarea limbajelor naturale.

Inteligența artificială se împarte în două clase:

Inteligență artificială tare (strong artificial intelligence) care se referă la domeniul în care calculatoarele pot gândi (cel puțin) la fel ca oamenii.

Inteligență artificială slabă (weak artificial intelligence) se referă doar la elemente de gândire (raționamente) adăugate calculatoarelor în scopul de a le face cât mai folositoare și acest lucru este deja realizat prin sisteme expert martor, comanda și conducerea automobilelor și programele de recunoaștere a vorbirii. Ce înseamnă a



Aplicații pentru Inteligența Artificială

Inteligența Artificială a intrat, sub mai multe forme, în viața noastră de zi cu zi. Există în magazinele online și e folosită ca să facă recomandări pentru noi cumpărături în funcție de achizițiile tale anterioare. E unul dintre motoarele inteligente din spatele unor platforme cum ar fi Siri și Alexa. IA analizează și recunoaște cine sau ce apare într-o fotografie, detectează spamul sau fraudele cu carduri bancare.

Pe lângă toate aceste utilizări, iată câteva dintre cele mai importante aplicații pentru inteligența artificială, unele dintre ele fiind deja comune în tehnologia de astăzi:

- ↪ în medicină
- ↪ în industria militară
- ↪ în finanțe
- ↪ în educație
- ↪ recunoaștere facială
- ↪ recunoaștere vocală
- ↪ manipulare foto și video

↪ creativitate artificială (scriere, compoziție muzicală etc.)

↪ procesarea limbajului natural

↪ recunoașterea scrisului de mână

↪ data mining

↪ realitatea virtuală

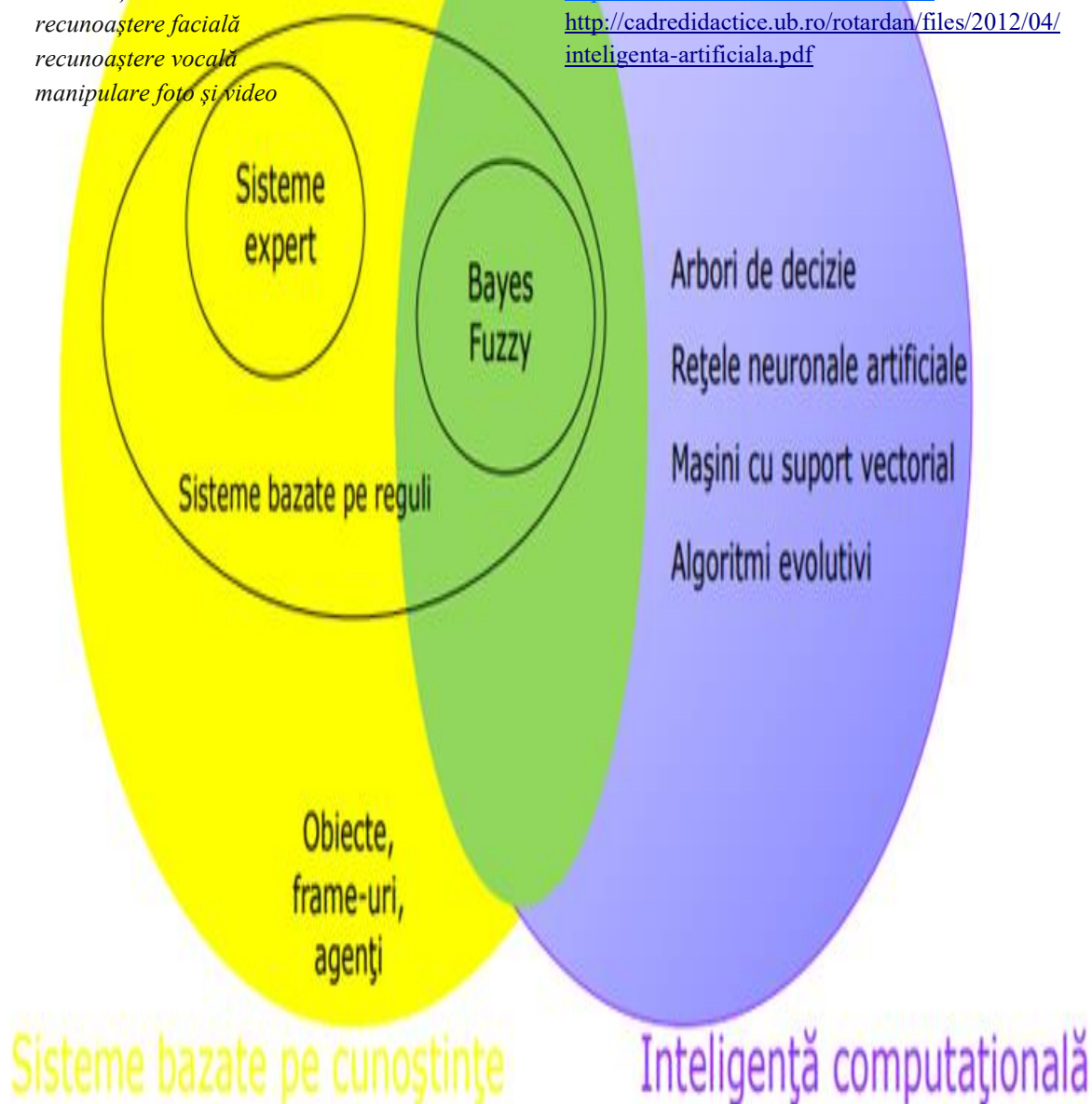
↪ procesarea imaginilor

În viitor computerele vor dispune de Inteligență Artificială, dar, în mod sigur, diferită de cea umană. La nivelul de cunoaștere actual, pentru oameni este aproape imposibil de imaginat ce formă de inteligență va avea entitatea virtuală aflată în memoria unui computer.

<https://gotech.world/inteligența-artificială-definiție-tipuri-de-ai-cum-invata-si-ce-aplicații-are/>

<http://www.catia.ro/articole/ai/ai.htm>

<http://cadredidactice.ub.ro/rotardan/files/2012/04/inteligența-artificială.pdf>



CONCURSURILE ȘCOLARE

Prof. Zeld Jenica, Liceul Teoretic Dunărea

Trăim într-o lume a ierarhizărilor și a competițiilor. Suntem îndemnați să urcăm pe trepte superioare de performanță și de competență. Ni se dă de înțeles că trebuie să

progresăm, să exploatăm la maximum ceea ce posedăm, să nu pierdem ocazia de a arăta ceea ce suntem în stare. Școala nu poate face abstracție de această tendință, ea însăși contribuind la generalizarea întrecerilor și la inducerea unui apetit al concurenței. Suntem învățați încă de mici să ne comparăm unii cu alții, să fim în fața altora, să ne autodepășim.

Despre concursurile școlare s-a discutat de-a lungul anilor, după 1989 acestea s-au dezvoltat ca număr, se desfășoară sub egida Ministerului Educației, dar și în parteneriate cu organizații non-profit, sau cu mediul privat. De multe ori concursurile sunt mai atractive din punct de vedere al recompenselor atunci când se implică mediul privat, în afara binecunoscutelor diplome, se acordă pentru elevii merituozii premii în bani sau obiecte.

Trebuie să spunem elevilor noștri care sunt avantajele participării la concursuri școlare și extrașcolare:

- Valorificarea interesului pentru un anumit domeniu

- Învățarea unor lucruri sau abilități noi.

- Învăț să faci față secolului.

- Creșterea stimei de sine.

- Concursurile școlare mai stimulează spiritul de inițiativă, abilitățile sociale și de comunicare, imaginația și le oferă o nouă oportunitate de a asimila cunoștințe noi.

Dezavantajele participării la concursuri

- Competițiile sunt o sursă de stres.

- Starea de epuizare fizică și psihică.

- Copilul dispune de mai puțin timp liber. Intervalul de timp acordat relaxării va fi mai redus datorită nevoii de a se pregăti pentru concurs.

- Relațiile sociale pot avea de suferit.

- Majoritatea concursurilor câștigă o bulina roșie datorită modului de desfășurare.

- Participarea la concursuri școlare înseamnă și absențe, ce-i drept - motivate, de la cursurile școlare, ceea ce înseamnă că la revenirea la școală, copilul va fi nevoit să muncească dublu la materiile la care a absentat.

Aș vrea să aduc însă în atenție cum profesorul ajunge să conducă spre înalta performanță elevii de la clasă. În primul rând trebuie să afirm că atunci când dorești să îți construiești o carieră în domeniul educației la un moment dat presiunea mediului școlar, și aici mă refer la colegi,

părinți, elevi etc. te determină să te implici în a aborda și concursurile școlare. Dacă ești profesor la o școală de prestigiu "neapărat" trebuie să participi încă din primii ani de activitate didactică.

Pentru elevii din clasele mici, părinții sunt foarte interesați în a-și trimite copiii la cât mai multe concursuri școlare. Motivele sunt diverse și bine întemeiate: nevoia de evaluare externă, (mai credibilă în condițiile în care învățătorii și profesorii pot fi câteodată prea îngăduitori), nevoia de diplome sau medalii pe care să le arătăm prietenilor și cu care copiii să se mândrească, diplome care demonstrează că ei sunt "mai buni" decât ceilalți.

Se cunoaște îngrijorarea părinților provocată de cunoașterea locului pe care sistemul educațional românesc îl are în topul sistemelor europene și de nevoia de a ști că propriul copil se va descurca și nu numai că se va descurca dar "va avea succes".

Mi-a fost destul de greu să explic părinților că succesul viitorului adult nu se măsoară în numărul diplomelor și nici măcar în mediile generale, ci în măsura în care acesta își identifică punctele forte (pasiunile dacă vreți) și reușește să își construiască o carieră în domeniul respectiv, măsura în care își dezvoltă abilități de leadership (public speaking, argumentare, gândire critică, perseverență), măsura în care reușește să se înțeleagă bine cu ceilalți (indiferent de religie, gen, naționalitate, culoare, dizabilități) și își contruiește relații de lungă durată și bazate pe valori, felul în care reușește să se bucure de viață și își găsește timp pentru călătorii, familie, prieteni și pasiuni.

Așadar am explicat atât părinților și copiilor că participarea la concursuri școlare trebuie să fie motivată în primul rând de bucuria interacțiunii sociale cu alți elevi de aceeași vârstă, de bucuria de a evada pentru scurt timp din mediul școlar și familial.

Toți elevii participanți au fost apreciați apoi în cadrul colectivului clasei, rolul meu a fost acela de a crește interesul și alor elevi pentru următoarele ediții ale diferitelor competiții.

Anul acesta școlar, în plină pandemie, copiii de clasa a VI-a au participat la un concurs denumit "Animații Scratch", în regim on-line, un concurs în cadrul căruia copiii, trebuiau să rezolve o aplicație pe o temă dată în timp de 24 ore (utilizând instrumente ale aplicației Scratch). După concurs, când am anunțat elevii care au luat premii, o mare parte din colegi nu au aplaudat sau nu i-au felicitat pe câștigători. Faptul că nu ne putem bucura de reușita altuia este o



Platforma educațională, un succes în “Școala online”

Profesor ing. Burlacu Cătălina Mercedes

Liceul Teoretic „Dunărea” din Galați

Evoluția alertă în domeniul educației din ultimul timp a determinat apariția și dezvoltarea a numeroase instrumente IT utile procesului didactic și implicit, dezvoltarea mai multor platforme de e-learning, care au devenit tot mai familiare profesorilor și elevilor. Învățarea prin e-learning poate fi definită ca o învățare la distanță într-un mediu educațional colaborativ, care îmbină metodele didactice tradiționale cu metode inovative bazate pe mijloace IT (computer, laptop, tabletă, smartphone) având ca obiectiv creșterea performanțelor individuale ale elevilor și atingerea unui rezultat. Învățarea prin e-learning se bazează pe o predare modernă, diferită de cea clasică, mult mai atractivă, în care un rol important îl are consolidarea cunoștințelor și evaluarea, realizate într-o manieră atractivă și adaptabilă nevoilor de învățare, atât a celor care dirijează învățarea, cât mai ales a elevilor.

O platformă de e-learning este un soft complex care permite administrarea unui domeniu (subdomeniu), gestionarea utilizatorilor pe domeniul respectiv, crearea și un management accesibil al cursurilor împreună cu activitățile și resursele asociate acestora, evaluarea online/offline sau autoevaluarea, comunicarea **sincronă** sau **asincronă** și multe altele.

La nivel mondial există un clasament al platformelor pentru învățare online, clasament care se alcătuiește pe baza statisticilor interne furnizate de aceste platforme, numărul de utilizatori înregistrați, țări în care se găsesc, suport disponibil în mai multe limbi, etc

Platformele de **e-learning** sunt un mediu de învățare *modular*, orientat pe obiect și dinamic ce permite oricui să contribuie la elaborare sau îmbunătățire, cu un sistem de management al și că un sistem de managementul învățării, un mediu virtual de învățare.

La nivel mondial platformele de **e-learning** sunt o comunitate ce oferă un mediu interactiv și integrat, centrat pe învățare și în acest context al învățării vor fi evidențiate în continuare *avantajele* care se remarcă din punctul de vedere al **profesorului**, dar și din cel al **elevului**.

Principiul de bază a unei platforme de **e-learning** este o învățare prin colaborare, bazată pe proiecte și sarcini individuale și de grup. Astfel, învățarea devine eficientă când profesorul construiește un material de învățare, pentru ca elevii săi să interacționeze cu acest material de învățare și să experimenteze, pentru a înțelege. În acest context, elevul este parte a unei comunități de învățare, în care este

pus în situația de a înțelege ce are de făcut, și de face împreună cu ceilalți colegi.

Platformele de **e-learning** oferă profesorilor o multitudine de instrumente utile în procesul didactic de predare – învățare – evaluare, o modalitate de definire a cursurilor pentru o adaptare la nevoile educabililor – elevi, sau alte categorii care trec printr-un proces de formare, fie inițială, fie continuă. Este important de spus că educabilii pot parcurge cursul **când, unde și cum** doresc, deoarece este adaptat necesităților lor, accesibil *oricând și oriunde*, chiar și în afara clasei, permițând colaborarea prin instrumente de comunicare moderne precum **forum, chat, blog sau wiki**.

Organizarea informațiilor este ierarhică: la bază se află cursurile care compun subcategoriile și care la rândul lor alcătuiesc categoriile de bază, accesul la cursuri fiind foarte ușor. Exact ca într-o bibliotecă, ale cărei rafturi sunt categoriile, fiecare raft conținând mai multe subcategorii, care pot fi disciplinele școlare – informatică, matematică, română, istorie etc., fiecare subcategorie/disciplină având în cuprins cursuri specifice: informatică pentru clasa a IX-a, pentru clasa a X-a etc.

Cursurile au două părți fundamentale: cele laterale care sunt formate din diverse blocuri cu diferite destinații, foarte utile în parcurgerea cursului și partea centrală, de o mai mare dimensiune în care sunt cuprinse activitățile și resursele cursului. Cursurile realizate pe o platformă de e-learning pot avea o copie de siguranță, care permite inserarea/restaurarea cursului pe o altă platformă, sau în cadrul unei alte categorii/subcategorii pe platforma pe care este realizat inițial. De exemplu, un curs de informatică al unui profesor poate fi utilizat de un alt profesor, fie în cadrul categoriei în care se găsește cursul inițial, fie în cadrul altei categorii, sau chiar pe o altă platformă.

Este evident faptul că la alegerea platformei de **e-learning** se are în vedere necesitățile grupului de formabili căruia se adresează, alegerea și folosirea acesteia bazându-se pe particularitățile și nevoile grupului, dar și a indivizilor care alcătuiesc acest grup. Important de menționat este caracteristica cu privire la numărul de utilizatori pe care au roluri diverse și modalități diferite de abordare ale conținutului și drepturile pe care aceștia le au, de la drepturi depline pe platformă, până la drepturi care permit doar vizualizarea unor conținuturi: administrator, manager, creator de curs, profesor, cursant/student/elev, vizitator.



Toți aceștia se conectează ca utilizatori autentificați în mod securizat, cu un nume de utilizator și parolă, ceea ce evidențiază încă un aspect important de controlat în ceea ce privește siguranța și securitatea datelor personale.

Foarte importante la o platformă sunt modalitățile de comunicare între utilizatori prin instrumente precum **forumul** (de uz general, de știri sau de tip întrebare-răspuns) și **wiki** – care permite elaborarea și accesarea unui conținut comun, acestea ca instrumente de comunicare asincronă/offline, dar și **chatul** ca instrument de comunicare **sincronă/online**. Evident că acestea permit o bună comunicare și colaborare între profesor și elevii săi, dar și între elevi. Există și posibilitatea de a se schimba mesaje prin intermediul **mesageriei** interne, dar și prin **email** și de a anunța utilizatorii cu privire la evenimentele viitoare.

Elaborarea unui curs de către profesor ține seama de particularitățile grupului căruia se adresează și deși profesorul este constrâns de un curriculum, el poate să abordeze întregul conținut al acestui curriculum adaptat nevoilor grupului supus instruirii. Reușita însușirii cunoștințelor dintr-un curriculum este asigurată de modalitatea de abordare, adaptată nivelului, nevoilor și particularităților membrilor unui grup, care poate să fie un grup de elevi, o clasă sau de ce nu, mai multe clase care vizează un conținut comun. Cursul se elaborează structurat pe teme, unități de învățare sau lecții astfel încât să respecte curriculumul, însă modalitatea prietenoasă de abordare oferă o atractivitate deosebită și ajută elevul să treacă de barierele unui conținut anost și neprietenos.

Accentul în cursurile elaborate în e-learning nu este pus pe furnizare de informații, ci pe **activitățile** care presupun schimb de idei și acumulare de cunoștințe noi bazate pe cunoștințe anterioare. **Clasa virtuală** sau **webinarul** este un alt instrument util, în care profesorul poate comunica **audio** și **video** cu elevii, în care sesiunile de lucru sunt online, putându-se desfășura și în afara clasei, învățarea fiind sincronă și dirijată de profesor. De menționat că există și modulul de autoinstruire, care permite elevului să parcurgă cursul asincron, în funcție de nevoia și timpul avut la dispoziție.

Activitățile realizate în platformele de e-learning sunt: **baza de date**, utilă în gestionarea informațiilor personale ale elevilor dintr-un grup sau clasă; **chestionarele** cu diferite întrebări care se pot adresa nu doar elevilor, ci și părinților care pot fi aplicate în timp real; **glosarul** creat în curs ca un instrument util și care se elaborează prin colaborare; **jocuri** desfășurate pe baza informațiilor conținute de un glosar; **lecții** cu pași impuși de condiții care să permită continuarea sau revenirea la un conținut care

trebuie revăzut; **feed-back** cu privire la activități desfășurate; **teme** care se pot da elevilor și care pot fi cu rezolvare online, offline sau cu încărcarea unor fișiere etc.

Resursele care se pot adăuga la un curs, sunt variate: cărți în format electronic, pagini cu conținut atractiv, adrese URL, galerie de imagini, fișiere de orice tip – documente, prezentări etc. Un rolul deosebit de important pe platformă îl are **evaluarea** care oferă o diversitate în abordare: modalitățile variate de elaborare a testelor, tipul atât de diversificat de întrebări puse la dispoziție de platformă și care se pot afișa în ordine aleatoare în test și cu date inițiale diferite pentru utilizatori diferiți, modalitățile de combinare și structurare a testelor, modalitățile de aplicare, care permit definirea timpului de lucru – durata testului, elevul știind cât timp mai are până la finalizare, data și ora când acest test poate fi susținut, aspectele legate de permisiunea de a repeta susținerea testului pentru îmbunătățirea rezultatelor, precum și diversele rapoarte pentru utilizatorii care au susținut testul și care se obțin după aplicarea unui test, de la analiza pe ansamblu, până la analiza pe itemi, cu obținerea unor diferite statistici și multe altele. Trebuie spus că în aceste teste corectarea este automată, fiind eliminat în acest fel subiectivismul care poate afecta nota acordată. Însă este important cum definește profesorul modalitatea de acordare a punctelor/notelor pe itemi raportat la ansamblul testului.

Referitor la activitatea de monitorizare și control a activităților desfășurate de utilizatori pe platformă, trebuie amintite toate rapoartele de activitate care se pot obține și care se pot utiliza activitățile de evaluare și anume rapoarte ale notelor, atât pentru fiecare utilizator (elev) sau pe ansamblu pentru toți utilizatorii înscriși la un curs, cât și pentru fiecare dintre itemii care compun testul pentru care se face raportarea.

Pe platformele de e-learning se pot integra softuri ca *Geogebra* și *Wiris*, utile profesorilor care predau matematică, dar și fizică sau chimie, *Hot Potatoes* (creare de teste sau jocuri de cuvinte încrucișate) și Audacity (editor audio), dar și altele precum *Sistem Antiplagiat*, utile la orice disciplină, completează cursurile create pentru matematică, fizică sau chimie cu elemente atractive, cu caracter științific, editate matematic și cu algoritmi ușor de elaborat..

Geogebra este un soft foarte cunoscut și utilizat pe scară largă, atât în algebră și geometrie, cât și în fizică sau chimie ce contribuie la o și mai mare atractivitate a cursurilor pentru disciplinele științifice – matematică în special, dar și fizică, chimie sau discipline tehnice.

Wiris ca soft integrat este util profesorilor care predau discipline precum matematica (în special) sau fizica și chimia. *Wiris* pune la dispoziția utilizatorilor 3



Prin combinarea acestor componente și integrare în platformă se pot obține cursuri atractive, complete care pot să acopere într-o proporție foarte mare cerințele curriculumului, cu o varietate mare de cerințe de învățare și cu o multitudine de instrumente utilizate în evaluare.

Avantajele pe care le oferă platforma se pot remarca doar în urma **elaborării, implementării și utilizării** cursurilor pentru orice disciplină școlară și în care se recomandă să fie integrate cât mai multe activități și resurse.

Un dezavantaj care trebuie menționat este **timpul** care trebuie alocat **creării și elaborării** cursurilor de către profesor, însă faptul că un curs odată elaborat poate fi utilizat ori de câte ori apare această necesitate, diminuează considerabil acest dezavantaj.

În învățământul preuniversitar un bun exemplu de utilizare și implementare a unei platforme este platforma Rețelei EDU de e-learning România. Activitățile acestei rețele sunt diverse, pe lângă cursurile pentru foarte multe discipline școlare sau cursurile de pregătire pentru examene naționale, existând multe alte activități: proiecte între școli,

conferințe online, concursuri, revista, festival de teatru, dar și cursuri de formare pentru profesori pe diverse paliere de interes.

Referințe:

- 1 <http://c4lpt.co.uk/top100tools/>
- 2 <https://de e-learning.org/>
- 3 <https://de e-learning.org/stats/>
- 4 <http://docs.de e-learning.org/dev/License>
- 5 <http://www.geogebra.org/cms/ro/>
- 6 <http://www.wiris.com/en/>
- 7 <https://edu.de e-learning.ro/>

Tag-uri



SUBIECTE ATESTAT PROGRAMARE
An școlar 2020-2021

Subiectul nr 1

Din fișierul **atestat.in** se citesc de pe prima linie un număr întreg $n \leq 100$ și apoi de pe a doua linie n numere întregi cu cel mult nouă cifre. Să se afișeze în ordine crescătoare toate numerele rotunde din fișier. Se numește număr **rotund** un număr care are aceleași cifre de 0 și de 1 în scrierea în baza 2.

Exemplu: numerele 10 și 1001 scrise în baza 2 sunt numere rotunde 10 în baza 2 reprezintă 2 în baza 10, iar 1001 în baza 2 reprezintă 9 în baza 10.

Dacă din fișierul **atestat.in** se citesc numerele:

10

14 9 35 27 78 2 10 11 17 36

Se vor afișa numerele 2 9 10 35

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
int rotund(int x)
{
    int c1=0, c0=0;
    while(x)
    {
        if(x%2==0)
            c0++;
        else
            c1++;
        x/=2;
    }
    return c1==c0;
}
int n, x, v[100], i, j, k;
ifstream f("atestat.in");
int main()
{
    f>>n;
    for(i=1; i<=n; i++)
    {
        f>>x;
        if(rotund(x)==1)
            v[++k]=x;
    }
    for(i=1; i<k; i++)
        for(j=i; j<=k; j++)
            if(v[i]>v[j])
                swap(v[i], v[j]);
    for(i=1; i<=k; i++)
        cout << v[i] << " ";
    return 0;
}
```



Subiectul nr 2

Din fișierul **atestat.in** se citesc numere întregi cu cel mult nouă cifre. Să se afișeze în ordine descrescătoare toate numerele de o cifră care nu au apărut în fișier. Dacă au apărut toate numerele de o cifră se va afișa mesajul **Au apărut toate cifrele**.

Exemplu: Dacă din fișierul **atestat.in** se citesc numerele:

123 45 2 67 3 2 13 1000 8 3 4 9 23 5 56
23 1 46789

Se va afișa secvența

7 6 0

Iar dacă din fișierul **atestat.in** se citesc numerele:

123 45 2 67 6 3 2 13 1000 8 145 0 21 7
3 4 9 23 5 56 23 1 46789

se va afișa mesajul **Au apărut toate cifrele**

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
int x, v[10], i, k;
ifstream f("atestat.in");
int main()
{
    while(f>>x)
        if(x<10)
            v[x]++;
    for(i=9;i>=0;i--)
        if(v[i]==0){cout << i
<< " "; k++;}
        if(k==0)
            cout<<"Au aparut toate
cifrele";
    return 0;
}
```

Subiectul nr 3

Din fișierul **atestat.in** se citesc de pe prima linie un număr întreg $n \leq 100$ și apoi de pe a doua linie n numere întregi cu cel mult nouă cifre. Să se afișeze în ordine crescătoare primele k numere dintre cele citite, unde k este un număr întreg citit de la tastatură.

Exemplu: Dacă din fișierul **atestat.in** se citesc numerele:

10

14 9 35 27 78 2 10 11 17 36, iar de la tastatură se citește numărul 5 **se vor afișa** numerele **2 9 10 11 14**

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
int n, x, v[100], i, j, k;
ifstream f("atestat.in");
int main()
{
    f>>n;
    for(i=1;i<=n;i++)
        f>>v[i];
    for(i=1;i<n;i++)
        for(j=i;j<=n;j++)
            if(v[i]>v[j])
                swap(v[i], v[j]);
    cin>>k;
    for(i=1;i<=k;i++)
        cout << v[i] << " ";
    return 0;
}
```



Subiectul nr 4

Din fișierul **atestat.in** se citesc numere întregi cu cel mult nouă cifre. Să se afișeze toate numerele prime citite. Dacă în fișier nu s-au citit numere prime se va afișa mesajul **NU EXISTĂ**.

Exemplu: Dacă din fișierul **atestat.in** se citesc numerele:

10 14 9 31 27 78 2 10 11 17 36 se vor afișa numerele 31 2 11 17, iar dacă din fișierul

atestat.in se citesc numerele

10 14 9 33 27 78 22 10 18 16 36 se va afișa mesajul **NU EXISTĂ**

Rezolvare:

```
#include
<iostream>
#include <fstream>
using namespace
std;
int x,k ;
ifstream f
```

```
("atestat.in");
int prim(int x)
{ for(int i=2;i<=x/2;i++)
  if(x%i==0)
    return 0;
  return 1;
}
int main()
{ while(f>>x)
  if(prim(x)==1)
  { cout<<x<<" ";
    k++;
  }
  if(k==0)
    cout<<"NU EXISTA";
  return 0;
}
```

Subiectul nr 5

Se citesc de la tastatură un număr întreg $n \leq 100$ și apoi n numere întregi cu cel mult nouă cifre.

Să se afișeze toate perechile de numere citite, care sunt prime între ele. Dacă nu s-au citit astfel de numere se va afișa mesajul **NU EXISTĂ**.

Exemplu: dacă pentru n se citește numărul 7 și apoi se citesc numerele 15 36 14 7 24 9 25 se vor afișa perechile (15, 14), (15, 7), (36, 7), (36, 25), (14, 9), (14,25), (7,24), (7, 9), (7, 25), (24, 25), (9, 25)

dacă pentru n se citește numărul 7 și apoi se citesc numerele 15 365 140 75 245 900 25 se va afișa mesajul **NU EXISTĂ**

Rezolvare:

```
#include <iostream>
using namespace std;
int cmmdc(int x, int y)
{ while(x!=y)
  {
    if(x>y)x-=y;
    else y-=x; }
}
```

```
return x==1;}
int n, v[100], i, j,k;
int main()
{ cin>>n;
  for(i=1;i<=n;i++)
    cin>>v[i];
  for(i=1;i<n;i++)
    for(j=i;j<=n;j++)
      if(cmmdc(v[i],v[j])==1)
        {cout<<"("<<v[i]
<<","<<v[j]<<") "; k++;}
  if(k==0) cout<<"NU
EXISTA";
  return 0;
}
```



Subiectul nr 6

Se citesc de la tastatură un număr întreg $n \leq 100$ și apoi n numere întregi cu cel mult nouă cifre. Să se afișeze ultimele două numere citite care conțin cel puțin o cifră de 5. Dacă nu s-au citit astfel de numere se va afișa mesajul **NU EXISTĂ**.

Exemplu: dacă pentru n se citește numărul 8 și apoi se citesc numerele 15 36 1545 327 284 2595 2567 87 se vor afișa numerele 2595 și 2567, iar dacă pentru n se citește numărul 7 și apoi se citesc numerele 123 366 134 78 24 9 289 se va afișa mesajul **NU EXISTĂ**.

Rezolvare:

```
#include <iostream>
using namespace
std;

int contine5(int x)
{ while(x!=0)
  { if(x%10==5)
    return
```

```
1;
  x/=10;}
return 0;
}
int n,x,i,p,u;
int main()
{
  cin>>n;
  for(i=1;i<=n;i++)
    {cin>>x;
     if(contine5(x)==1)
      {p=u;
       u=x;}}
  if(p==0||u==0)
  cout<<"NU EXISTA";
  else
  cout<<p<<" "<<u;
  return 0;
}
```

Subiectul nr 7

Se citesc de la tastatură un număr întreg $n \leq 100$ și apoi n numere întregi cu cel mult nouă cifre. Să se afișeze toate numerele situate între poziția celui mai mic număr citit și poziția celui mai mare număr citit.

Exemplu: dacă pentru n se citește numărul 7 și apoi se citesc numerele 15 7 14 24 36 9 25 se vor afișa numerele 14 24, iar dacă pentru n se citește numărul 7 și apoi se citesc numerele 123 366 134 78 24 9 289 se vor afișa numerele 134 78 24.

Rezolvare:

```
#include
<iostream>
#include <climits>
using namespace
std;
int n, x, v[100],
i, Min=INT_MAX,
Max=INT_MIN, pMin,
pMax;
int main()
```

```
{
  cin>>n;
  for(i=1;i<=n;i++)
    {cin>>v[i];
     if(v[i]<Min) {Min=v
[i]; pMin=i;}
     if(v[i]>Max) {Max=v
[i]; pMax=i;}
    }
  if(pMin>pMax)
    swap(pMin,
pMax);
  for(i=pMin+1;i<pMax;i++)
    cout << v[i] << " ";
  return 0;
}
```



Subiectul nr 8

Din fişierul **atestat.in** se citesc numere întregi cu cel mult nouă cifre. Să se afişeze toate numerele palindrom citite, situate după primul număr par. Dacă în fişier nu s-au citit numere palindrom sau nu s-a găsit niciun număr palindrom după primul număr par, se va afişa mesajul **NU EXISTĂ**.

Exemplu:

Dacă din fişierul **atestat.in** se citesc numerele:

10

14341 191 312 27 828 2 10 121 17 3 se vor afişa numerele 828 2 121 3, iar dacă din

fişierul **atestat.in** se citesc numerele

10

14 91 371 27 78 625 120 18 16 36 se va afişa mesajul **NU EXISTĂ**

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
int n, x, v[100], i, j, k, p=0;
ifstream f("atestat.in");
int pal(int x)
{   int inv=0, x1=x;;
    while(x)
        {   inv=inv*10+x%10; x/=10;   }
    return x1==inv; }
int main()
{
    f>>n;
    for(i=1;i<=n;i++)
        {f>>v[i];
         if(v[i]%2==0)
             if(p==0)p=i;}
    for(i=1;i<=n;i++)
        if(pal(v[i])==1&& i>p)
            {   cout << v[i]
<< " "; k++; }
        if(k==0) cout<<"NU
EXISTA";
    return 0; }
```

Subiectul nr 9

Din fişierul **atestat.in** se citesc de pe prima linie un număr întreg $n \leq 100$ şi apoi

de pe a doua linie n numere întregi cu cel mult nouă cifre. Să se afişeze toate numerele care sunt numere divizibile cu 10, după eliminarea cifrelor impare.

Exemplu:

Dacă din fişierul **atestat.in** se citesc numerele:

10

14032 92005 375 27003 78 2 10 1081

1720 36 se vor afişa numerele 92005

27003 1720

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
int n, v[100], i;
ifstream f("atestat.in");
int div10(int x)
{
    int y=0, p=1;
    while(x)
        {if(x%2==0)
            {
                y=y+x%10*p; p*=10;
            }
            x/=10;
        }
    return y!=0&&y%10==0; }
int main()
{
    f>>n;
    for(i=1;i<=n;i++)
        {f>>v[i];
         if(div10(v[i])==1)
             cout << v[i] << " ";}
    return 0;
}
```



Subiectul nr 10

Din fişierul **atestat.in** se citesc numere întregi cu cel mult nouă cifre. Să se afişeze în ordine crescătoare, o singură dată, numerele de două cifre care au apărut în fişier.

Exemplu:

Dacă din fişierul **atestat.in** se citesc numerele:
123 45 2 67 3 2 13 1000 8 67 67 3 4 9
23 5 56 23 1 46789

Se vor afişa numerele: 13 23 45 56 67

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
int x, v[100], i;
```

```
ifstream f("atestat.in");
int main()
{
    while(f>>x)
        if(x>=10&&x<=99)
            v[x]++;
    for(i=10;i<=99;i++)
        if(v[i]>0)cout << i<<" ";
    return 0; }
```

Subiectul nr 11

Din fişierul **atestat.in** se citesc numere întregi cu cel mult nouă cifre. Să se afişeze în ordinea inversă citirii toate numerele echilibrate. Se numeşte număr **echilibrat**, un număr care are în scrierea lui, un număr de cifre pare egal cu numărul de cifre impare. Presupunem că se va citi cel puţin un număr echilibrat.

Exemplu: Dacă din fişierul **atestat.in** se citesc numerele:

123 45 2345 67 3 2 13 14 8 34 9 23 5
5276 2134 1 46789

Se vor afişa numerele: 2134 5276 34 14 67
2345 45

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
int x, v[100], i, n;
ifstream f("atestat.in");
int echilibrat(int x)
{
    int par=0, impar=0;
```

```
while(x)
{
    if(x%2==1)impar++;
    else par++;
    x/=10;
}
return impar==par;
}
int main()
{
    while(f>>x)
        v[++i]=x;
    n=i;
    for(i=n;i>=1;i--)
        if(echilibrat(v[i])
==1)cout << v[i]<<" ";
    return 0;
}
```



Subiectul nr 12

Se citesc de la tastatură un număr întreg $n \leq 100$ și apoi n numere întregi cu cel mult nouă cifre. Să se afișeze toate numerele pare, care sunt situate între poziția primului număr prim și poziția ultimului număr prim. Dacă nu s-au citit numere prime se vor afișa toate numerele citite.

Exemplu: dacă pentru n se citește numărul 12 și apoi se citesc numerele 15 36 7 14 24 9 25 13 192 19 204 234 se vor afișa numerele 14 24 192, iar dacă citește numărul 12 și apoi se citesc numerele 15 36 74 14 24 9 2525 1326 192 196 204 234 se vor afișa numerele 15 36 74 14 24 9 2525 1326 192 196 204 234

Rezolvare:

```
#include <iostream>
using namespace
std;
int x, v[100], i,
n, p1, p2;
int prim(int x)
{ for(int
i=2; i<=x/2; i++)
```

```
if(x%i==0)
return 0;
return 1;
}
int main()
{
cin>>n;
for(i=1; i<=n; i++)
{ cin>>v[i];
if(prim(v[i])==1) if
(p1==0) p1=i; else p2=i;
}
if(p1!=0&& p2!=0)
{for(i=p1+1; i<p2; i++)
if(v[i]%2==0) cout << v
[i]<<" ";}
else
{for(i=1; i<=n; i++)
cout << v[i]<<" ";}
return 0;
}
```

Subiectul nr 13

Din fișierul **atestat.in** se citesc numere întregi cu cel mult nouă cifre. Să se afișeze toate numerele **perfecte** în ordine inversă citirii. Un număr este **perfect** dacă este egal cu suma divizorilor săi proprii, inclusiv 1. Dacă în fișier nu s-au citit numere perfecte se va afișa mesajul **NU EXISTĂ**.

Exemplu: Dacă din fișierul **atestat.in** se citesc numerele:

10

14 6 31 496 78 2 28 11 17 36 se vor afișa numerele 28 496 6, iar dacă din fișierul

atestat.in se citesc numerele

10

14 9 33 27 78 22 10 18 16 36 se va afișa mesajul **NU EXISTĂ**

Rezolvare:

```
#include <iostream>
#include <fstream>
```

```
using namespace std;
int k, v[100], i, n;
int perfect(int x)
{int s=0;
for(int i=2; i<=x/2; i++)
if(x%i==0)
s=s+i;
return s+1==x;
}
ifstream f("atestat.in");
int main()
{
while(f>>v[++i])
n=i;
for(i=n; i>=1; i--)
if(perfect(v[i])){ cout
<< v[i]<<" "; k++;}
if(k==0) cout<<"NU EXISTA";
return 0;
}
```



Subiectul nr 14

Se consideră un tablou bidimensional cu **m** linii și **n** coloane. Să se afișeze elementele tabloului astfel încât liniile să devină coloane. **Exemplu:**

m=2, n=3 7 6 3 se va afișa 7 1
 1 9 6
 6 9
 3 6

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
ifstream f("matrice.in");
int main()
{
    int a[10][20], n, m, i, j;
    f >> m >> n;
    for(i=1; i<=m; i++)
        for(j=1; j<=n; j++)
```

```
        f >> a[i][j];
    for(i=1; i<=n; i++)
    {
        for(j=1; j<=m; j++)
            cout << a[j][i] << " ";
        cout << endl;
    }
    return 0;
}
```

Subiectul nr 15

Se consideră un tablou bidimensional cu **n** linii și **n** coloane, afișati cea mai mare componentă și pozițiile pe care le ocupă. **Exemplu:** n=3

3 9 8 se va afișa max=9 pozitii
 1 2
 5 7 7
 3 1
 9 0 9
 3 3

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace
std;
ifstream f
("matrice.in");
int main()
{
    int a[10]
[20], n, i, j;
    f >> n;
    for(i=1; i<=n;
i++)
        for(j=1;
j<=n; j++)
            f >> a[i][j];
    int max1=a[1][1];
```

```
    for(i=1; i<=n; i++)
        for(j=1; j<=n; j++)
            if(max1<a[i][j])
                max1=a[i][j];
    cout << max1 << endl;
    for(i=1; i<=n; i++)
        for(j=1; j<=n; j++)
            if(max1==a[i][j])
                cout << i << "
" << j << endl;
    cout << endl;
    return 0;
}
```



Subiectul nr 16

Se consideră un tablou bidimensional cu **n** linii și **n** coloane care este utilizat ca o tablă de șah. Se citesc două valori **x** și **y** care vor reprezintă poziția unui cal pe tabla de șah. Să se afișeze coordonatele pozițiilor din tablou pe care poate fi mutat calul.

Exemplu: $x=3$ $y=1$ se va afișa: 2 3 4 3 1
2 5 2.

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace
std;
ifstream f
("matrice.in");
int main()
{
    int a[10]
[20],n,i,j,x,y;
    f>>n>>x>>y;
    if(y+2<=n)
    {
        if(x-1>=1)
            cout<<x
-1<<" "<<y+2<<endl;
        if(x+1<=n)
```

```
            cout<<x+1<<"
"<<y+2<<endl;
    }
    if(y-2>=1)
    {
        if(x-1>=1)
            cout<<x-1<<" "<<y-
2<<endl;
        if(x+1<=n)
            cout<<x+1<<" "<<y-
2<<endl;
    }
    if(x+2<=n)
    {
        if(y-1>=1)
            cout<<x+2<<" "<<y-
1<<endl;
        if(y+1<=n)
            cout<<x+2<<"
"<<y+1<<endl;
    }
    if(x-2>=1)
    {
        if(y-1>=1)
            cout<<x-2<<" "<<y-
1<<endl;
        if(y+1<=n)
            cout<<x-2<<"
"<<y+1<<endl;
    }
    return 0;
}
```

Subiectul nr 17

Se utilizează un tablou bidimensional cu **m** linii și **n** coloane pentru a reține notele obținute de către **m** elevi la **n** materii. Afișați media pentru fiecare elev.

Exemplu: $m=3$ $n=2$ 8 6 se afișează: 7
6 8.5

Rezolvare:

```
#include
<iostream>
#include <fstream>
using namespace
std;
ifstream f
("matrice.in");
int main()
{float med;
    int a[10]
[20],n,i,m,j;
```

```
f>>m>>n;
    for(i=1; i<=m; i++)
        for(j=1; j<=n; j++)
            f>>a[i][j];
    for(i=1; i<=m; i++){med=0;
        for(j=1; j<=n; j++)
            med+=a[i][j];
        med=med/n;
        cout<<i<<"
"<<med<<endl;
    }
    return 0;
}
```



Subiectul nr 18

Scrieți un program care citește m și n reprezentând numărul de linii și coloane ale unui tablou bidimensional. Programul afișează suma componentelor de pe marginea tabloului.

Exemplu: $m=3$ $n=4$ 1 2 1 0 se afișează :
 $s=21$.

1 7 0 8

0 1 3 4

Rezolvare:

```
#include
<iostream>
#include <fstream>
using namespace
std;
```

```
ifstream f
("matrice.in");
int main()
{int s=0, a[10]
[20],n,i,m,j;
f>>m>>n;
for(i=1; i<=m; i++)
for(j=1; j<=n; j++)
f>>a[i][j];
for(i=1; i<=m; i++)
s=s+a[i][1]+a[i][n];
for(j=2; j<=n-1; j++)
s=s+a[1][j]+a[m][j];
cout<<s<<endl;
return 0;
}
```

Subiectul nr 19

Scrieți un program care prin operații de interschimbare de linii într-un tablou bidimensional cu m linii și n coloane să ordoneze crescător elementele de pe coloana c .

Exemplu: $m=3$ $n=3$ $c=1$ 6 2 9 se afișează: 1
9 3

1 9 3

5 5 0

5 5 0

6 2 9

Rezolvare:

```
#include <iostream>
#include <fstream>
using namespace std;
ifstream f("matrice.in");
int main()
{ int s=0, a[10]
[20],n,i,m,j,c,k;
f>>m>>n>>c;
for(i=1; i<=m; i++)
for(j=1; j<=n; j++)
f>>a[i][j];
for(i=1; i<=m-1; i++)
for(j=i+1; j<=m; j++)
if(a[i][1]>a[j]
[1])
for(k=1; k<=n;
```

```
k++)
swap(a[i]
[k],a[j][k]);
for(i=1; i<=m; i++)
{
for(j=1; j<=n; j++)
cout<<a[i][j]<<" ";
cout<<endl;
}
return 0;
}
```



Subiectul nr 20

Scrieți un program care inserează într-un tablou bidimensional cu **m** linii și **n** coloane, pe poziția **l** (linie în matrice) un tablou unidimensional cu **n** elemente.

Exemplu: m=3 n=3 l=2 6 2 9 9 9 2

se afișează: 1 9 3

1 9 3 9 9 2

5 5 0

1 9 3

5 5 0

Rezolvare:

```
#include
<iostream>
#include <fstream>
using namespace
std;
ifstream f
("matrice.in");
int main()
{
    int s=0;
    int a[10]
```

```
[20],n,i,m,j,l,v
[20];
f>>m>>n>>l;
for(i=1; i<=m;
i++)
    for(j=1;
j<=n; j++)
        f>>a
[i][j];
for(i=1; i<=n;
i++)
f>>v[i];
```

```
for(i=m; i>=1; i--)
for(j=1; j<=n; j++)
a[i+1][j]=a[i][j];
for(j=1; j<=n; j++)
a[l][j]=v[j];
m++;
for(i=1; i<=m; i++)
{
    for(j=1; j<=n; j++)
        cout<<a[i][j]<<"
";
    cout<<endl;
}
return 0;
}
```

Subiectul nr 21

Se consideră două șiruri de caractere **s1**, **s2** (litere mici și spații) fiecare cu cel mult 50 de caractere. Să se afișeze numărul de apariții al șirului **s2** în șirul **s1**.

Exemplu: s1=Ana are mere si are si pere
s2=are se va afișa 2.

Rezolvare:

```
#include
<iostream>
#include <cstring>
using namespace
std;
int main()
{char s1[51],s2
[51],*p;
int nrap=0;
cin.get(s1,51);
cin.get();
cin.get(s2,51);
p=strstr(s1,s2);
while(p)
{
```

```
nrap++;
p=strstr(p+1,s2);
}
cout << nrap
<< endl;
return 0;
}
```



<p>Subiectul nr 22</p> <p>Se citește un șir de caractere s cu maxim 70 de caractere (litere mici și spații). Codificați șirul dat afișând în locul fiecărei litere codul ei ASCII. Sa se afișeze șirul codificat, cu un spațiu între coduri și câte o steluță pentru fiecare spațiu din text.</p> <p>Exemplu: s= Am un mar se afișează: 65 109*117 110**109 97 114.</p> <p>Rezolvare:</p> <pre> #include <iostream> #include <cstring> using namespace std; int main() {char s[51]; int i; cin.get(s,51); for(i=0;i<strlen </pre>	<pre> (s);i++) if(s[i]==' ') cout << '*'; else if(s[i+1]!=' ') cout<<int(s[i]) <<' '; else cout<<int(s [i]); return 0; } </pre>
<p>Subiectul nr 23</p> <p>Se citește un șir de caractere s format numai din cifre. Afișați șirul cu cifrele grupate câte trei, de la dreapta la stânga, grupele fiind separate de punct.</p> <p>Exemplu: s=52364879523641652 se afișează: 52.364.879.523.641.652</p> <p>Rezolvare:</p> <pre> #include <iostream> #include <cstring> using namespace std; int main() {char s[51],s1 [256]; int i,k=0,j=0; </pre>	<pre> cin.get(s,51); for(i=strlen(s)- 1;i>=0;i--) {s1[j++]=s[i]; k++; if(k%3==0 && i! =0) s1[j++]='.'; } s1[j]=NULL; strcpy(s, strrev (s1)); cout<<s; return 0; } </pre>
<p>Subiectul nr 24</p> <p>Se citește un șir de caractere s. Construiți un șir s1 format din caracterele din șirul s mai puțin cifrele.</p> <p>Exemplu: s= Ana are 3 mere si 7 pere se afișează: Ana are mere si pere</p> <p>Rezolvare:</p> <pre> #include <iostream> #include <cstring> using namespace std; int main() {char s[51],s1 [256]; </pre>	<pre> int i,j=0; cin.get(s,51); for(i=0;i<=strlen (s)-1;i++) { if(isdigit(s [i])==0) s1[j++]=s [i]; } s1[j]=NULL; cout<<s1; return 0; } </pre>



Subiectul nr 25

Se citește un șir de caractere **s** (numai litere mici). Modificați șirul **s** astfel încât toate vocalele să fie la începutul șirului iar consoanele la sfârșit.

Exemplu: **s=vsaedhue** se afișează:**aeuvsdh**
(soluție posibilă)

Rezolvare:

```
#include <iostream>
#include <cstring>
using namespace
std;
int main()
{char s[51];
int i,k=0;
cin.get(s,51);
for(i=0;i<=strlen
(s)-1;i++)
{    if(strchr
```

```
("aeiou",s[i])!=0)
    {swap(s[k],s
[i]); k++;}
}
cout<<s;
return 0;
}
```

Subiectul nr 26

Se citesc **n** șiruri de caractere fiecare cu maxim 50 de caractere. Un șir conține numele și prenumele unui elev separate printr-un spațiu. Să se afișeze șirurile în ordine alfabetică doar pentru elevii al căror nume începe cu o vocală.

Exemplu: **n=4** se afișează
Avram Dan
Popescu Vasile
Ene Ioana
Ene Ioana
Avram Dan
Dan Amalia

Rezolvare:

```
#include <iostream>
#include <cstring>
#include <fstream>
using namespace
std;
ifstream f
("sir.in");
int main()
{    char s[51],cuv
[10][51];
    int
n,i,k=0,j=0;
    f>>n;
    for( i=1; i<=n;
i++)
    {    f.getline
```

```
(s,51); f.get();
        if(strchr
("AEIOU",s[0])!=0)
            strcpy
(cuv[++k],s);
    }
    for( i=1; i<=k-
1; i++)
        for(j=i+1;
j<=k; j++)
            if
(strcmp(cuv[i],cuv
[j])>0)
                swap(cuv[i],cuv
[j]);
    for( i=1; i<=k;
i++) cout<<cuv[i]
<<endl;
    return 0;
}
```



Subiectul nr 27

Se citesc șiruri de caractere **s1**, **s2** fiecare cu maxim 50 de caractere (litere mici și spații). Se cere să se afișeze cuvintele comune celor două șiruri.

Exemplu: s1=am toate timbrele din seria 1986
s2=1984 este seria din care nu am tot se afișează: am din seria (o soluție posibilă)

Rezolvare:

```
#include <iostream>
#include <cstring>
#include <fstream>
using namespace std;
ifstream f("sir.in");
int main()
{   char s1[51],s2[51],cuv1
    [10][51],cuv2[10][51],*p;
    int k1=0,k2=0,i,j;
    f.getline(s1,51);
```

```
f.get();
    f.getline(s2,51);
f.get();
    p=strtok(s1," ");
    while(p)
    {   strcpy(cuv1[++k1],p);
        p=strtok(NULL," ");
    }
    p=strtok(s2," ");
    while(p)
    {   strcpy(cuv2[++k2],p);
        p=strtok(NULL," ");
    }
    for( i=1; i<=k1; i++)
    {for(j=1; j<=k2; j++)
        if(strcmp(cuv1[i],cuv2
[j])==0)
            cout<<cuv1[i]
<<endl;
    }
    return 0;
```

Subiectul nr 28

Se dă o matrice pătratică de dimensiune **n**, cu elemente numere naturale. Să se afișeze indicii liniilor pentru care suma elementelor este număr par.

Se va defini și folosi subprogramul **suma**, cu 3 parametri:

x – matricea

n – dimensiunea matricei

p – un număr natural, $1 \leq p \leq n$

care va returna suma elementelor de pe linia **p** a matricei **x**.

Exemplu: n=4 și 1 5 9 2 se afișează 2, 3, 4.

```
3 6 2 1
2 0 6 8
1 3 8 6
```

Rezolvare:

```
#include <iostream>
using namespace std;
int suma(int x[100][100],int
n,int p){
    int s=0,j;
    for(j=1;j<=n;j++)
        s+=x[p][j];
    return s;
}
int main()
```

```
{   int x[100][100],n,p,i,j;
    cin>>n;
    for(i=1;i<=n;i++)
        for(j=1;j<=n;j++)
            cin>>x[i][j];
    for(p=1;p<=n;p++)
        if(suma(x,n,p)%2==0)
            cout<<p<<endl;
}
```



Subiectul nr 29

Se dă o matrice cu **n** linii și **m** coloane. Să se determine câte dintre elementele situate pe linii cu indice număr par sunt prime. Se va defini o funcție *prim* care verifică dacă un număr este sau nu prim.

Exemplu:

Pentru $n=4$, $m=3$ și matricea:

2	35	9
7	17	25
24	6	5
13	28	2

Se va afișa 4.

Rezolvare:

```
#include <iostream>
using namespace std;
int a[100][100],n,m,i,j,nr;
int prim(int x)
```

```
{ for(int i=2; i<=x/2; i++)
    if(x%i==0)
        return 0;
    return 1;
}
int main()
{ cin>>n>>m;
  for(i=1; i<=n; i++)
    for(j=1; j<=m; j++)
      cin>>a[i][j];
  nr=0;
  for(i=2; i<=n; i=i+2)
    for(j=1; j<=m; j++)
      if(prim(a[i][j])==1)
        nr++;
  cout<<nr;
  return 0;
}
```

Subiectul nr 30

Se dă un vector cu **n** elemente, numere naturale. Afișați în ordine descrescătoare valorile din vector care sunt prime cu ultimul element al vectorului. Se va defini o funcție care să determine cel mai mare divizor comun a două numere naturale.

Exemplu: Pentru $n=7$ și vectorul **12 35 47 26 287 30 5** se va afișa 287 47 26 12

Rezolvare:

```
#include <iostream>
using namespace std;
int n,i,v[100],aux;
void citire(int &n,int v[100])
{ int i;
  cin>>n;
  for(i=1; i<=n; i++)
    cin>>v[i];
}
void ordonare(int v[100],int n)
{
  int i,j;
  for(i=1; i<=n-1; i++)
    for(j=i+1; j<=n; j++)
      if(v[i]<v[j])
        swap(v[i],v
[j]);
}
int cmmdc(int a,int b)
```

```
{
  while(a!=b)
    if(a>b)
      a=a-b;
    else
      b=b-a;
  return a;
}
int main()
{
  citire(n,v);
  aux=v[n];
  ordonare(v,n-1);
  for(i=1; i<=n; i++)
    if(cmmdc(v[i],aux)==1)
      cout<<v[i]<<" ";
  return 0;
}
```



Subiectul nr 31

Se dă o matrice cu **n** linii și **n** coloane. Calculați cel mai mare divizor comun al elementelor aflate deasupra diagonalei principale. Se va utiliza o funcție care să determine cel mai mare divizor comun a două numere.

Exemplu: Pentru matricea patrată de dimensiune $n=4$ și elementele:

2 45 75 300

5 15 45 30

10 95 7 450

2 5 67 10

Se va afișa 15.

Rezolvare:

```
#include <iostream>
using namespace std;
int cmmdc(int a, int b)
```

```
{
    while (a!=b)
        if (a>b) a-=b;
        else b-=a;
}
int main()
{
    int a[10][10],d,n,i,j;
    cin>>n;
    for(i=1; i<=n; i++)
        for(j=1; j<=n; j++)
            cin>>a[i][j];

    d=a[1][2];
    for(i=1; i<=n; i++)
        for(j=i+1; j<=n; j++)
            d=cmmdc(a[i][j],d);
    cout<<d<<" ";
    return 0;
}
```

Subiectul nr 32

Se dau coordonatele carteziene a **n** puncte în plan. Să se determine distanța maximă dintre un punct dat și originea sistemului de coordonate și numărul de puncte situate la acea distanță față de origine. Se va folosi un subprogram care să determine distanța dintre un punct și origine.

Exemplu: Pentru $n=7$ și punctele: (2, 5) (4, 9) (5, 4) (2, 9) (6, 8) (8, 4) (3, 9)

Se va afișa 10 și 1.

Rezolvare:

```
#include <iostream>
#include <cmath>
using namespace std;
struct punct
{
    int a,b;
} v[100];
int n,i, nr=0;
float dist(punct x)
{
    float c=sqrt
(x.a*x.a+x.b*x.b);
    return c;
}
int main()
{
    float maxi=-1;
    cin>>n;
```

```
for(i=1; i<=n; i++)
{
    cin>>v[i].a>>v[i].b;
}
for(i=1; i<=n; i++)
    if(dist(v[i])>maxi)
    {
        maxi=dist(v[i]);
        nr=1;
    }
    else if(dist(v[i])
==maxi)
        nr++;
    cout<<maxi<<" "<<nr;
    return 0;
}
```



Subiectul nr 33

Se dă un vector cu **n** elemente, numere naturale distincte. Ordonăți crescător elementele situate înaintea valorii maxime din vector și descrescător elementele situate după această valoare. Se va folosi un subprogram care să determine indicele valorii maxime din vector.

Exemplu: Pentru $n=7$ și vectorul 3 26 14 38 23 27 1 se va afișa 3 14 26 38 27 23 1.

Rezolvare:

```
#include <iostream>
using namespace std;

int v[100], n, i, j, maxi, b,
aux;
void citire(int &n, int v[100])
{
    int i;
    cin>>n;
    for(i=1; i<=n; i++)
        cin>>v[i];
}
int maxim(int n, int v[100])
{
    int i;
    int maxi=-9999;
    for(i=1; i<=n; i++)
        if(v[i]>maxi)
            maxi=v[i];
}
```

```
for(i=1; i<=n; i++)
    if(v[i]==maxi)
        return i;
}
void crescator (int v[100],int b)
{
    int i,j;
    for(i=1; i<=b-1; i++)
        for(j=i+1; j<=b; j++)
            if(v[i]>v[j])
                swap(v[i],v[j]);
}
void descrescator (int v[100],int
b,int n)
{
    int i,j;
    for(i=b; i<=n-1; i++)
        for(j=i+1; j<=n; j++)
            if(v[i]<v[j])
                swap(v[i],v[j]);
}
int main()
{
    citire(n,v);
    b=maxim(n,v);
    crescator(v,b);
    descrescator(v,b,n);
    for(i=1; i<=n; i++)
        cout<<v[i]<<" ";
    return 0;
}
```

Subiectul nr 34

Se dau **n** numere naturale nenule. Ordonăți descrescător cele **n** numere după numărul lor de divizori.

Exemplu: $n=7$ și 12 4 8 33 2 5 24 se va afișa 24 12 33 8 4 5 2

Rezolvare:

```
#include <iostream>
using namespace std;
int n,i,j,k,g,h;
struct diviz
{
    int nr,div;
};
diviz v[100];
int divizor(int x)
{
    int d;
    d=2;
```

```
for(j=2; j<=x/2; j++)
    if(x%j==0) d++;
return d;
}
int main()
{
    cin>>n;
    for(i=1; i<=n; i++)
    {
        cin>>v[i].nr;
        v[i].div=divizor(v[i].nr);
    }
    for(i=1; i<n; i++)
        for(j=i+1; j<=n; j++)
            if (v[i].div < v
[j].div)
                swap(v[i],v[j]);
    for(i=1; i<=n; i++)
        cout<<v[i].nr<<" ";
    return 0;
}
```



Subiectul nr 35

Se dă un vector cu n elemente, numere naturale. Afișați în ordine crescătoare valorile prime din vector. Se va defini o funcție *prim* care verifică dacă un număr este sau nu prim.

Exemplu: $n=7$ și vectorul 5 24 13 8 6 19 7 se va afișa: 5 7 13 19.

Rezolvare:

```
#include <iostream>
using namespace std;
int i,j,n,v[50];
void citire()
{   int i;
    cin>>n;
    for(i=1; i<=n; i++) cin>>v
[i];
}
void ordonare(int v[50],int n)
{
    int i,j;
```

```
    for(i=1; i<=n-1; i++)
        for(j=i+1; j<=n; j++)
            if(v[i]>v[j]) swap
(v[i],v[j]);
}
int prim(int x)
{   int j;
    for(j=2; j<=x/2; j++)
        if(x%j==0) return 0;
    return 1;
}
int main()
{   citire(); ordonare(v,n);
    for(i=1; i<=n-1; i++)
        if(prim(v[i]))
            cout<<v[i]<<"
";
    return 0;
}
```

Subiectul nr 36

Din fișierul **date.in** se citesc numere naturale. Calculați și afișați pe ecran câte dintre numerele citite au toate cifrele identice. Se va folosi o funcție care primește ca parametru un număr natural n și returnează 1 dacă n are toate cifrele identice și 0 în caz contrar.

Exemplu: Dacă fișierul conține numerele 12221 34563 33 44444 99999 6555 3453 7 se va afișa 4.

Rezolvare:

```
#include <fstream>
#include <iostream>
using namespace std;
ifstream fin("date.in");
int cifreegale(int n)
{   int c=n%10;
    while(n)
    {   if(n%10!=c) return 0;
        n=n/10;
```

```
    }
    return 1; }
int main()
{   int n,k=0;
    while(fin>>n)
        if(cifreegale(n)) k++;
    cout<<k;
    fin.close();
    return 0;
}
```



Subiectul nr 37

Scrieți un program în C++, care să citească de la tastatură un număr natural **n** și să afișeze pe ecran suma factorialelor cifrelor lui. Se va utiliza o funcție care să returneze factorialul unui număr natural transmis ca parametru.

Exemplu: n=3652 se va afișa: 848 (6+720+120+2=848).

Rezolvare:

```
#include <iostream>
using namespace std;
int n, c;
long s;
long factorial(int x)
```

```
{ int p=1;
  for(int i=1; i<=x; i++)
  p*=i;
  return p;
}
int main()
{ cin>>n;
  s=0;
  while (n)
  { c=n%10;
    s+=factorial(c);
    n=n/10;
  }
  cout<<s;
  return 0;
}
```

Subiectul nr 38

Despre fiecare dintre cei **n** elevi dintr-o clasă se citesc din fișierul **elev.in** următoarele date:
- numele (șir de caractere de lungime maxima 20)
- prenumele (șir de caractere de lungime maxima 20)

- media notelor în oral la informatică
- nota la teza de informatică

Să se afișeze, ordonați alfabetic, elevii corigenți la informatică. Pentru fiecare elev corigent se vor afișa: numele, prenumele și media la informatică.

Rezolvare:

```
#include <fstream>
#include <iostream>
#include <string.h>
using namespace std;
struct elev
{ char nume[30], pren[30];
  float mo;
  int teza;
  float media; };
ifstream f("elev.in");
elev c[100];
void citire(elev &e)
{ f>>e.nume; f>>e.pren;
  f>>e.mo; f>>e.teza;
  e.media=(e.mo*3+e.teza)/4;
}
void afisare(elev e)
{ cout<<e.nume<<"
  "<<e.pren<<" "<<e.media<<endl;
}
void ordonare(int n,elev c
```

```
[100])
{ int i,gata;
  elev aux;
  do
  { gata=1;
    for (i=1; i<n; i++)
      if (strcmp(c
[i].nume,c[i+1].nume)>0)
        { aux=c[i]; c[i]
=c[i+1];
          c[i+1]=aux;
        }
    }while (!gata);
}
int main ()
{ int n,i;
  f>>n;
  for (i=1; i<=n; i++) citire
(c[i]);
  ordonare (n,c);
  for (i=1; i<=n; i++)
    if (c[i].media<4.5)
      afisare(c[i]);
  return 0;
}
```



Subiectul nr 39

Numerele **puternice** sunt acele numere care au numărul divizorilor mai mare decât numărul divizorilor oricărui număr natural mai mic decât numărul respectiv.

Să se afișeze al **n**-lea număr puternic. Se va folosi o funcție pentru a verifica dacă un număr este puternic sau nu.

Exemplu: Pentru $n=5$ se va afișa 12. În șirul numerelor 1 2 3 4 5 6 7 8 9 10 11 12

Rezolvare:

```
#include <iostream>
using namespace std;
int n, k, a;
int divizori (int x)
{   int nr=0;
    for (int i=1; i<=x; i++)
        if (x%i==0) nr++;
```

```
        return nr;
    }
int puternic(int x)
{   int d;
    d=divizori(x);
    for (int i=1; i<x; i++)
        if (divizori(i)>=d)
            return 0;
    return 1;
}
int main()
{   cin>>n;
    k=1; a=1;
    while(k<n)
    {   a++;
        if (puternic(a)) k++;
    }
    cout<<a;
    return 0;
}
```

Subiectul nr 40

Se dau înălțimile a **n** copii, numerotați de la 1 la **n**, exprimate prin numere naturale. Afișați numerele de ordine ale copiilor în ordinea crescătoare a înălțimii lor.

Rezolvare:

```
#include <iostream>
using namespace std;
struct copil
{
    int nr;
    float h;
};
copil a[100];
int n, i, j;
void citire()
{
    cin>>n;
    for(i=1; i<=n; i++)
    {
```

```
        a[i].nr=i;
        cin>>a[i].h;
    }
}
int main()
{
    citire();
    for(i=1; i<n; i++)
        for(j=i+1; j<=n; j++)
            if(a[i].h>a[j].h)
                swap(a[i],a
[j]);
    for(i=1; i<=n; i++)
        cout<<a[i].nr<<"
";
    return 0;
}
```



Baza de date Presa

Conține următoarele tabele:

Reviste (*idr* number(3), *titlu* varchar2(30), *pret* number(5,2), *categorie* varchar2(30))

Abonati (*ida* number(4), *nume* varchar2(30), *adresa* varchar2(50))

Abonamente (*idr* number(3), *ida* number(4), *data_ab* date)

Următoarele opt probleme fac referire la baza de date **Presa**

Subiectul nr. 1

Să se afișeze suma prețurilor revistelor.

Să se afișeze titlurile revistelor la care este abonat Popescu Ion.

Subiectul nr. 2

Afișați revistele în ordinea alfabetică a titlurilor.

Să se afișeze revistele care au prețul mai mic decât prețul mediu.

Subiectul nr. 3

Care sunt revistele cu prețul minim?

Afișați numele abonaților la reviste din categoria SF.

Subiectul nr. 4

Afișați numele abonaților care stau pe strada Brăilei.

Să se afișeze numele abonaților și titlurile revistelor la care s-a făcut abonament în luna curentă.

Subiectul nr. 5

Afișați titlurile revistelor care încep cu litera A.

Care este suma prețurilor revistelor la care s-a abonat Popescu Ion?

Subiectul nr. 6

Afișați prețul mediu al revistelor din categoria Știință și tehnică.

Câți abonați au făcut abonamente în primele 6 luni din an?

Subiectul nr. 7

Să se afișeze prețul mediu pentru fiecare categorie de reviste.

Să se afișeze, crescător după data la care s-a făcut abonamentul, toate abonamentele din baza de date.

Subiectul nr. 8

Să se afișeze revistele grupate pe categorii, iar cele din aceeași categorie – crescător după preț.

Să se afișeze numele persoanelor care au făcut abonament la aceeași categorie de reviste ca cele la care s-a abonat Popescu Ion.



Tabela Angajati:

Angajati (*id number(3), nume varchar2(30), data_nas date, data_ang date, salariu number(5)*)

Următoarele trei probleme fac referire la tabela Angajati:

Subiectul nr. 9

Să se afișeze angajații cu un salariu peste medie.

Să se afișeze numele celui mai tânăr angajat.

Subiectul nr. 10

Să se afișeze salariile mărite cu 10% pentru angajații care își serbează ziua de naștere în luna curentă.

Să se afișeze angajații în ordinea crescătoare a salariilor, iar cei cu același salariu – în ordine alfabetică.

Subiectul nr. 11

Să se afișeze numele și salariile angajaților care s-au angajat în aceeași lună cu luna de naștere.

Care este suma salariilor angajaților care au un salariu peste medie?

Baza de date Biblioteca conține următoarele tabele:

carti (*id number(3), titlu varchar2(40), autor varchar2(30), editura varchar2(30), an_aparitie number(4), pret number(5)*)

cititori (*idc number(4), nume varchar2(10), data_n date, loc varchar2(20), mail varchar2(30)*)

imprumut (*id number(3), idc number(4), data_impr date*)

Următoarele cinci probleme fac referire la baza de date Biblioteca

Subiectul nr. 12

Să se afișeze numele cititorilor care au împrumutat cărți în anul 2020.

Să se afișeze informațiile despre cărțile cu prețul mai mare decât prețul mediu.

Subiectul nr. 13

Să se afișeze numărul de autori de carte (dacă un autor a scris mai multe cărți, se va număra o singură dată).

Să se afișeze numele cititorilor care au împrumutat cea mai scumpă carte.

Subiectul nr. 14

Să se afișeze valoarea cărților apărute în anul 2000.

Să se afișeze cititorii din Galați care au împrumutat cărți de Mihai Eminescu.

Subiectul nr. 15

Să se afișeze numele cititorilor care au adresă de mail pe yahoo.ro.

Să se afișeze prețul cărților împrumutate de Popescu Ion.

Subiectul nr. 16

Să se afișeze numele cititorilor, descrescător după data la care au împrumutat cărți.

Să se afișeze numele cititorilor care au împrumutat cărți în lunile de primăvară, toamnă și iarnă.



Baza de date Magazin conține următoarele tabele:

Produse (*idp number(3), denumire varchar2(30), pret number(5,2), cantitate number(4), categorie varchar2(30)*)

Furnizori (*idf number(4), nume varchar2(30), localitate varchar2(50)*)

Comenzi (*idpnumber(3), idf number(4), data_com date*)

Următoarele patru probleme fac referire la baza de date Magazin

Subiectul nr. 17

Să se afișeze toți furnizorii care nu sunt din Galați.

Din ce localități sunt furnizorii pentru care azi s-au cumpărat produse din categoria *sport*?

Subiectul nr. 18

Să se afișeze numărul de produse cu stoc > 10.

Să se afișeze datele la care s-au comandat produse cu preț maxim.

Subiectul nr. 19

Să se afișeze prețul mediu al produselor din categoria *sport*.

Să se afișeze valoarea totală a produselor furnizate de firme din Galați.

Subiectul nr. 20

Să se afișeze valoarea totală a produselor care au prețul cuprins între 20 și 30 lei.

Baza de date Cinema conține următoarele tabele:

Filme (*idf number(3), nume varchar2(40), gen varchar2(20), buget number(9), anaparitie number(4), limba varchar2(40)*)

Actori (*ida number(3), nume varchar2(40), datanastere Date, tara varchar2(40)*)

Distribuție (*idf number(3), ida number(3)*)

Următoarele șapte probleme fac referire la baza de date Cinema

Subiectul nr. 21

Să se afișeze filmele difuzate în limba engleză ordonate după gen iar în cadrul aceluiași gen vor fi ordonate după buget descrescător

Să se afișeze în ordine alfabetică actorii care au jucat în filme în anul 2019

Subiectul nr. 22

Să se determine câte filme apărute în 2019 sunt în baza de date

Să se afișeze în ordine alfabetică actorii americani care au jucat în filme

Subiectul nr. 23

Să se afișeze care a fost cel mai mare buget alocat pentru un film apărut în anul 2019

Să se afișeze ordonați după vârstă actorii care au jucat în filme apărute în anul 2018



Subiectul nr. 24

Să se afișeze câți actori americani sunt în baza de date
Să se afișeze actorii născuți în 1980 care au jucat în filme

Subiectul nr. 25

Să se afișeze filmele ordonate crescător după anul apariției. Filmele apărute în același an se vor ordona alfabetic după nume
Să se determine câți actori au jucat în filmele din 2019 difuzate în limba engleză

Subiectul nr. 26

Să se determine bugetul mediu pentru fiecare categorie de filme
Să se afișeze în ordine alfabetică actorii americani care au jucat în filme din seria **Star Wars**

Subiectul nr. 27

Să se afișeze filmul/filmele care au avut cel mai mic buget în 2019
Să se afișeze cel mai tânăr actor care a fost distribuit în filmul *Gravity*

Baza de date Clinica conține următoarele tabele:

Pacienti (*idp number(3), nume varchar2(40), afectiune varchar2(40)*)

Medicamente (*idm number(3), denumire varchar2(50), categorie varchar2(20), pret number(8,2)*)

Tratamente (*idp number(3), idm number(3)*)

Următoarele șase probleme fac referire la baza de date Clinica

Subiectul nr. 28

Să se afișeze în ordine alfabetică pacienții și afecțiunile lor
Să se determine costul tratamentului pentru pacientul **Avram Ion**.

Subiectul nr. 29

Să se afișeze în ordine alfabetică denumirile medicamentelor și prețurile lor
Să se afișeze numele pacienților și denumirile medicamentelor care fac parte din tratamentul lor

Subiectul nr. 30

Să se afișeze prețul mediu pentru medicamentele din categoria *analgezice*
Să se afișeze pacienții și medicamentele lor din categoria *antitermice*

Subiectul nr. 31

Să se afișeze medicamentele grupate pe categorii iar cele din aceeași categorie descrescător după preț
Să se afișeze numele pacienților care primesc analgezice

Subiectul nr. 32

Să se afișeze prețul mediu pentru fiecare categorie de medicamente
Să se afișeze costul medicației fiecărui pacient



Subiectul nr. 33

Să se afișeze medicamentele descrescător după preț. Medicamentele cu același preț vor fi ordonate alfabetic.

Să se înlocuiască numele pacientului *Avram Mihai* cu *Avramescu Mihai*

Baza de date *Închirieri Auto* conține următoarele tabele:

clienti (*idc number(3)*, *nume varchar2(40)*, *adresa varchar2(40)*)

automobile (*ida number(3)*, *numar_inmatriculare varchar2(10)*, *marca varchar2(20)*, *an_fabricatie number(4)*)

inchirieri (*idc number(3)*, *ida number(3)*, *data_in Date*, *numar_zile number(3)*)

Următoarele șapte probleme fac referire la baza de date *Închirieri Auto*

Subiectul nr. 34

Să se determine câte mașini sunt în baza de date

Să se afișeze numele clienților cu adresa de ***Galați*** care au închiriat mașini și mărcile mașinilor închiriate

Subiectul nr. 35

Să se afișeze alfabetic mărcile automobilelor. Mașinile cu aceeași marcă vor fi ordonate descrescător după anul fabricației

Să se afișeze numele clienților, mărcile mașinilor închiriate și numărul de zile închiriate. Datele vor fi ordonate după numele clienților.

Subiectul nr. 36

Să se determine câte mașini înmatriculate în ***Galați*** sunt în baza de date

Să se afișeze numele clientului și marca pentru autoturismul închiriat care are numărul de înmatriculare ***GL-05-GEO***

Subiectul nr. 37

Să se afișeze datele mașinilor fabricate în 2019 ordonate alfabetic după marcă

Să se afișeze datele tuturor mașinilor închiriate de ***Dima Ion***: număr de înmatriculare, marca și numărul de zile închiriate.

Subiectul nr. 38

Să se afișeze numărul de mașini din fiecare an de fabricație, ordonate după anul de fabricație

Să se actualizeze cu valoarea ***GL-03-IDA*** numărul numărul de înmatriculare pentru automobilul cu numărul de înmatriculare ***GL-03-ADI***.

Subiectul nr. 39

Să se determine câte zile au fost închiriate mașinile din baza de date

Să se elimine din baza de date autoturismul cu numărul de înmatriculare ***GL-03-ABC***

Subiectul nr. 40

Să se afișeze datele clienților ordonate după nume pentru clienții care au adresa în ***Galați*** sau ***Brăila***

Să se afișeze clienții, numărul de înmatriculare și marca automobilului pentru închirieri care presupun mai mult de 10 zile



București, 17.04.2018

Către Asociația Profesorilor de Informatică „Tudor Sorin” Galați

Vă anunțăm ca publicația pe care o editați a fost înregistrată și a primit codul de identificare ISSN, după cum urmează:

Revista informatică (Im)puls++ (Online) =
ISSN 2601 - 5951, ISSN-L 2067 - 9815

Stimați colegi și elevi, vă invităm să colaborați la revista noastră. Responsabilitatea în legătură cu originalitatea și conținutul științific al articolelor, problemelor aparține în exclusivitate autorilor.

Felicități elevilor și profesorilor care au înțeles că scopul acestei reviste este de a crea o relație elev-profesor fructuoasă. Așteptăm opiniile, problemele rezolvate și propuse pe adresa de mail a revistei noastre: revistainformaticagalati@yahoo.com

Colectivul de redacție:

- Redactor șef: **CĂTĂLINA BURLACU** – Liceul Teoretic „Dunărea”
- Redactor șef adjunct **GEORGETA-IULIA BALACEA, C. N. V. A.**
- Redactori principali profesorii:
 - Inspector școlar prof. Mușunoiu Novetschi Iuliana Monica
 - Stan Maria, Liceul Tehnologic "Radu Negru"
 - Joc Genia, C.N. A.I.C





(iM)PULS++ DE LA A LA Z:

Ambiție	Bunavoință	Curaj
Dorință	Energie	Forță
Geniu	Hotărâre	Intelligență
Joc	Knowledge	Luptă
Muncă	Nădejde	Optimism
Putere	Qualité	Reușită
Succes	Stiință	Tehnologie
Tel	Utilizare	Voință
Www	Xilografie	Youth
	Zel	