

**PROGRAMA ȘCOLARĂ
PENTRU OPȚIONALUL**

MICUL PROGRAMATOR

Clasele a III-a și a IV-a

Profesor coordonator,

NOTĂ DE PREZENTARE

Opționalul Micul programator face parte din aria curriculară Matematică și Științele naturii, cu durata de 1 an școlar, 1 oră pe săptămână. În cadrul acestui opțional elevii vor descoperi programarea și utilizarea calculatorului prin platforme interactive și atractive. Cu ajutorul platformei SCRATCH vor învăța să programeze, folosind blocuri de cod și vor realiza animații, jocuri sau povești. În cadrul acestui opțional elevii își vor dezvolta creativitatea, gândirea algoritmică și abilitățile de lucru pe calculator.

Programele Scratch sunt construite din blocuri grafice, codul programului fiind mai ușor de citit și împărtășit decât în cazul altor limbaje de programare (colaborare). Să pornești de la o idee și să te gândești cum să realizezi un program în Scratch necesită perseverență și practică (autoorganizare).

În timp ce lucrează la proiecte Scratch, elevii învață să selecteze, creeze și să administreze diferite forme de media, printre care text, imagine, animație și sunet (informații). Pe măsură ce elevii câștigă experiență în crearea de media, ei devin mai receptivi și mai critici când trebuie să analizeze media din jurul lor. Comunicarea eficientă în lumea de astăzi necesită mai mult decât abilitatea de a citi și scrie text (comunicare în masă). Scratch angajează elevii în alegerea, manipularea și integrarea unei varietăți de media pentru a se exprima creativ și persuasiv (comunicare). Pentru a construi proiecte, elevii trebuie să coordoneze în timp interacțiunile dintre mai mulți actori (gândire sistemică). Crearea unui proiect Scratch constă în generarea unei idei, transformarea ei într-o serie de pași și implementarea cu blocurile de programare Scratch (rezolvare de probleme). Elevii vor învăța nu doar cum să rezolve o problemă predefinită, ci sunt pregătiți să găsească soluții noi atunci când apar noi provocări (creativitate).

Există multe instrumente diferite care pot fi folosite pentru informatica creativă. La acest opțional vom folosi **Scratch**, adică un limbaj de programare gratuit, disponibil pe <http://scratch.mit.edu>. Cu Scratch se poate crea o mare varietate de proiecte media interactive - animații, povești, jocuri și multe altele - ce pot fi partajate într-o comunitate online. De la lansarea Scratch în mai 2007, sute de mii de oameni din întreaga lume au creat și împărtășit mai mult de 6 milioane de proiecte.

COMPETENȚE GENERALE

1. Utilizarea responsabilă și eficientă a tehnicii de calcul și de comunicații
2. Rezolvarea unor probleme elementare prin construirea unor algoritmi de prelucrare a informației
3. Elaborarea creativă de produse informatice care să valorifice conexiunile dintre disciplina Informatică și TIC și societate

COMPETENȚE SPECIFICE ȘI EXEMPLE DE ACTIVITĂȚI DE ÎNVĂȚARE

Competențe specifice	Exemple de activități de învățare
1. Aplicarea operațiilor specifice editoarelor grafice în vederea realizării unor produse informatice	<ul style="list-style-type: none">• realizarea și utilizarea unor personaje grafice pentru ilustrarea unei povești• realizarea unei felicitări, a unui afiș publicitar, a unui logo etc.
2. Utilizarea unui mediu grafic interactiv pentru exersarea algoritmilor.	<ul style="list-style-type: none">• rularea unor algoritmi model, modificarea lor conform unor cerințe și crearea unor algoritmi noi, prin utilizarea instrumentelor specifice• identificarea efectelor aplicării unor algoritmi cu valențe practice sau de joc• rezolvarea unor probleme-joc prin aplicarea unor instrumente interactive specifice
3. Aplicarea etapelor de rezolvare pentru cerințe simple, corespunzătoare unor situații familiare.	<ul style="list-style-type: none">• discutarea unei cerințe de prelucrare (tema prezentării/scopul animației/cerința algoritmică) și identificarea rezultatelor așteptate deduse din aceasta• stabilirea unor strategii de rezolvare și alegerea uneia pentru rezolvarea unei teme.• exerciții de urmărire pas cu pas, modificare, completare, restructurare a unui algoritm pentru a obține un algoritm cu o cerință dată• construirea, pentru o cerință dată, a unui algoritm ce utilizează în mod repetat o secvență de prelucrare
4. Implementarea unui algoritm într-un mediu grafic interactiv.	<ul style="list-style-type: none">• utilizarea unui mediu interactiv care permite implementarea folosind elemente grafice (de exemplu, Scratch) pentru a crea aplicații simple prin operații de

Competențe specifice	Exemple de activități de învățare
	<p>mișcare, sunete, vizualizare text etc.</p> <ul style="list-style-type: none"> • analizarea exemplurilor existente pe Internet specifice mediului grafic selectat și modificarea acestora pentru a îndeplini alte funcțiuni • participarea la un concurs de aplicații realizate individual sau în echipă, utilizând un mediu grafic
<p>5. Manifestarea creativă prin utilizarea unor aplicații simple de construire a unor jocuri digitale</p>	<ul style="list-style-type: none"> • participarea la evenimente de tip “Hour of Code” • implicarea în activități colaborative utilizând aplicațiile studiate (de exemplu, participarea la un joc didactic de echipă, crearea în echipă a unui joc educațional/povești etc.) • analizarea codului unui joc simplu în scopul identificării modului de realizare a funcționalității acestuia, modificarea codului pentru a obține alte efecte și analiza comparativă a efectelor obținute printr-un schimb liber de idei

CONȚINUTURI

DOMENII DE CONȚINUT	CONȚINUTURI
1. Editoare grafice	Rolul unui editor grafic
	Elemente de interfață specifice
	Crearea, deschiderea și salvarea fișierelor grafice
	Comenzi pentru selectare, copiere, mutare, ștergere
	Redimensionarea, trunchierea, rotația unei imagini
	Instrumente de desenare
	Utilizarea culorilor în prelucrarea imaginilor; crearea culorilor personalizate
2. Utilizarea mediului de programare SCRATCH	Noțiuni introductive
	Scurtă introducere în limbajul SCRATCH – descriere, prezentare
	Elemente de interfață specifice mediului grafic interactive
	Mișcare și sunet – prezentarea meniului și aplicații simple
	Evenimente și control – introducere, aplicații
	Să ne jucăm cu culorile – prezentare, opțiuni și aplicații
	Proiect de evaluare finală
3. Prezentarea instrumentelor din SCRATCH care pot fi folosite pentru a rezolva probleme elementare	Crearea propriilor personaje – utilizarea meniului de design pentru personaje, upload imagini, desenare.
	Sunete, voci și muzică – prezentare, opțiuni, aplicații
	Crearea animațiilor – aplicarea unor situații simple, familiare în animații, povești, jocuri.
	Crearea poveștilor – utilizarea instrumentelor pentru a crea povești, după algoritmi simple
	Identificarea efectelor aplicării unor algoritmi cu valențe practice sau de joc
4. Elaborarea creativă de produse informatice care să valorifice conexiunile dintre disciplina Informatică și TIC și societate	Implementarea algoritmilor care conțin structura secvențială în mediul grafic. Implementarea algoritmilor care conțin structura alternativă în mediul grafic
	Rularea unor algoritmi model, modificarea lor conform unor cerințe și crearea unor algoritmi noi, prin utilizarea instrumentelor specifice
	Produse informatice create prin utilizarea elementelor specifice mediului grafic interactiv
	Analizarea codului unui joc simplu în scopul identificării modului de realizare a funcționalității acestuia, modificarea codului pentru a obține alte efecte și analiza comparativă a efectelor obținute printr-un schimb liber de idei

SUGESTII METODOLOGICE

Instruirea se desfășoară într-un laborator de informatică în care – pentru optimizarea demersului didactic – este recomandat să existe un număr de stații de lucru egal cu numărul elevilor din clasă, conectate în rețea și cu posibilitate de acces la serviciile Internet necesare pentru formarea competențelor cuprinse în programă.

Configurația calculatoarelor trebuie să permită rularea aplicațiilor selectate de profesor dintre cele recomandate în programă. În laborator se recomandă să existe dispozitive periferice și de stocare a informației (de exemplu, o imprimantă, un memory-stick, boxe etc.). În activitatea frontală cu elevii, utilizarea unui videoproiector focalizează atenția acestora asupra materialelor suport, eficientizând procesul de învățare. Organizarea laboratorului trebuie să permită atât desfășurarea momentelor de lecție în care se introduc noi cunoștințe, precum și a momentelor de lecție în care se formează deprinderi practice, pe calculator.

Lucrul efectiv pe calculator este inițiat și coordonat de profesor, prin formularea clară a sarcinilor de lucru, cu menționarea bugetului de timp alocat și a criteriilor de evaluare. În funcție de specificul clasei și de particularitățile colectivului de elevi, profesorul va adapta nivelul de complexitate a sarcinilor de lucru.

Specificul disciplinei implică utilizarea unor metode didactice active. Se recomandă îmbinarea metodelor clasice (de exemplu: demonstrația, problematizarea, algoritmizarea) cu metodele moderne (de exemplu: învățarea prin descoperire, observarea sistematică, proiectul, portofoliul, studiul de caz, jocul de rol).

La începutul studiului unui anumit software este recomandat să le fie prezentat elevilor un produs demonstrativ, realizat cu software-ul respectiv. Analiza unor exemple de bună practică va crea o imagine de ansamblu asupra facilităților oferite de software-ul respectiv și va motiva elevii în demersul de realizare a unor produse de înaltă calitate.

În abordarea aplicațiilor interdisciplinare, se recomandă corelarea activităților didactice cu nivelul achizițiilor dobândite de elevi la disciplinele vizate.

Prezentarea la clasă a unor algoritmi simpli, cunoscuți, reprezentați în diverse limbaje de programare, are doar un rol informativ, elevul fiind invitat să observe numai similitudinile cu pseudocodul. În același timp, cunoscând finalitatea utilizării algoritmilor în informatică, programarea și diversitatea limbajelor de programare, elevul va fi motivat să construiască algoritmi pentru rezolvarea problemelor, pentru ca ulterior să îi poată "traduce" în limbaj de programare.

S recomandă implicarea elevilor în activități colective de tip *Hour of Code*, care să permită formarea și dezvoltarea competențelor specifice.

Pentru stimularea lucrului în echipă, se recomandă realizarea unor proiecte pe o temă dată, pe parcursul a 2-3 ore și prezentarea în fața colectivului de elevi a produselor realizate, încurajând procesul de autoevaluare.

Pentru competența generală 1:

- se pot utiliza filme didactice existente pe YouTube, simulatoare virtuale, tutoriale, software educațional disponibil online sau pe diverse platforme educaționale;
- pentru editare grafică se pot utiliza aplicații specifice sistemului de operare utilizat sau resurse gratuite existente online;

Pentru competența generală 2:

- se pot utiliza medii grafice online sau aplicații care se instalează pe calculatorul de lucru, cum ar fi Scratch

- pentru animații grafice se poate utiliza aplicația Scratch;
- se recomandă ca noțiunea de algoritm să fie introdusă pornind de la exemple concrete, familiare elevilor, fie din viața reală, fie de la alte discipline școlare;
- instrucțiunile grafice specifice mediului ales pot fi descoperite rezolvând sarcini de lucru atractive, de tip joc.
- exercițiile algoritmice pot să vizeze: mișcarea unui personaj pe o traiectorie dată (Scratch), construirea unor figuri geometrice și a unor desene formate din figuri geometrice - de exemplu să construiască un pătrat repetând de patru ori o deplasare înainte și o întoarcere la stânga etc.;
- profesorul va utiliza cu precădere exerciții algoritmice (rulări pas cu pas) și jocuri algoritmice;
- se pot alege prelucrări algoritmice cu valențe de joc sau cu valențe practice, din ariile de interes și de înțelegere ale elevilor (specifice vârstei și spațiului socio-cultural în care trăiesc);

Pentru competența generală 3:

- se recomandă lucrul pe proiecte cu teme interdisciplinare, care să stimuleze creativitatea elevilor; de exemplu, proiectarea unui joc educativ în care elevii să exerseze tabla adunării sau tabla înmulțirii sau chiar sarcini de lucru mai complexe; crearea unei povești și transpunerea acesteia în format digital, crearea unui joc, realizarea unui colaj de imagini cu colegii de clasă sau de la activitățile școlare și extrașcolare ale clasei, crearea unor personaje sau modificarea unor personaje și integrarea acestora în proiecte mai complexe etc.
- studiul fiecărei aplicații va începe cu o secvență de lecție demonstrativă care să pună în evidență atractivitatea produselor, diversitatea instrumentelor de lucru și ușurința în utilizare a aplicației.

Sugestii de modalități de evaluare

- probe practice, utilizate în vederea evaluării gradului de stăpânire a priceperilor și deprinderilor de utilizare a unui sistem de calcul pentru îndeplinirea sarcinilor proprii și pentru relaxare;
- realizarea unui proiect.

BIBLIOGRAFIE

- Agape, M. (n.d.). *Să învățăm programarea jucându-ne în Scratch - Curs pentru începători*. Orșova: 2016.
- Karen Brennan, C. B. (n.d.). *Informatică creativă*. Harvard Graduate School of Education.
- Raluca Elena MATEI, M. T. (2015). *Introducere în Scratch*. București: EUROGAMA INVENT.
- <http://shallwelearn.com/blog/stories/23343953-Should-We-Learn-Scratch-Programming-eBook.pdf>
- http://wiki.scratch.mit.edu/wiki/Scratch_3. <http://info.scratch.mit.edu/Scratch2FAQ>
- http://info.scratch.mit.edu/Support/Scratch_Cards
- <http://ccdmures.ro/cmsmadesimple/pdf/jakab.pdf>
- <http://cymplecy.wordpress.com/scratchgpi>