



Ministerul Educației al Republicii Moldova
Centrul de Excelență în Informatică și Tehnologii Informaționale



"Aprob"

Directorul Centrului de Excelență în
Informatică și Tehnologii Informaționale

 Vitalie Zavadschi

20 decembrie 2016

Curriculumul modular
S.O.O.018 Limbaje de programare

Specialitatea: 61110 Calculatoare
Calificarea: Tehnician pentru suportul tehnic al calculatoarelor

Chișinău 2016

Curriculumul a fost elaborat în cadrul Proiectului *EuropeAid/133700/C/SER/MD/12*
"Asistență tehnică pentru domeniul învățământ și formare profesională
în Republica Moldova",
implementat cu suportul financiar al Uniunii Europene



Autori:

Arcan Petru, grad didactic superior, Centrul de Excelență în Informatică și Tehnologii Informaționale.

Sineavschi Lucia, grad didactic doi, Centrul de Excelență în Energetică și Electronică.

Aprobat de:

Consiliul metodic-științific al Centrului de Excelență în Informatică și Tehnologii Informaționale.



Director

Vitalie Zavadschi

20 decembrie 2016

Recenzenți:

1. GAMA COMPUTER SRL/NEURON, adresa: str. V.Alecsandri 1, MD-2009 mun. Chișinău
Director: Mincheivici Sergiu
2. VIC-COM INFO SRL, adresa: str. Albișoara 68/3 of. 72, MD, mun. Chișinău, Director:
Tabuci Victor

Adresa Curriculumului în Internet:

Portalul național al învățământului profesional tehnic

<http://www.ipt.md/ro/produse-educationale>.

Cuprins

I. Preliminarii	4
II. Motivația, utilitatea modulului pentru dezvoltarea profesională.....	4
III. Competențele profesionale specifice modulului	4
IV. Administrarea modulului	4
V. Unitățile de învățare	5
VI. Repartizarea orientativă a orelor pe unități de învățare	9
VII. Studiu individual ghidat de profesor	9
VIII. Lucrările de laborator recomandate	10
IX. Sugestii metodologice	10
X. Sugestii de evaluare a competențelor profesionale	12
XI. Resursele necesare pentru desfășurarea procesului de studii	16
XII. Resursele didactice recomandate elevilor	16

I. Preliminarii

Importanța studierii unui limbaj de programare reiese din principala funcție a lui, cea de mijloc de interacțiune practică dintre om și calculator. Există multe limbaje de programare. Ele oferă posibilitatea de a transpune algoritmi de rezolvare a problemelor din limbajul uman în cel al calculatorului. Cu apariția mediilor de programare integrate este posibilă utilizarea limbajelor de programare de către diferite categorii de utilizatori. Sunt rezolvate o mulțime de probleme din multe domenii de activitate umană.

Modulul respectiv este structurat în șase unități de învățare și este destinat formării competențelor de realizare efectivă a aplicațiilor într-un limbaj de programare. Unitățile de curs ce trebuie studiate până la demararea procesului de instruire la acest modul sunt:

- G.01.O.001 Tehnologia informației.
- F.06.O.015 Arhitectura calculatoarelor

II. Motivația, utilitatea modului pentru dezvoltarea profesională

Rolul modului "Limbaje de programare" constă în formarea competențelor și abilităților necesare pentru îndeplinirea sarcinilor de lucru în procesul de asistență la elaborarea aplicațiilor.

La finalizarea modului, elevul va fi capabil să:

- descrie structura interfeței mediului de programare integrat;
- realizeze algoritmul de elaborare a problemei;
- realizeze etapele de rezolvare a problemei;
- rezolve probleme concrete cu aplicarea structurilor de date;
- rezolve probleme cu utilizarea fișierelor;
- rezolve probleme cu utilizarea claselor și obiectelor;
- interprete pas cu pas îndeplinirea programului.

III. Competențele profesionale specifice modului

- CS1. Utilizarea elementelor de interfață ale mediului de programare integrat;
- CS2. Identificarea etapelor de realizare a unui program;
- CS3. Aplicarea elementelor de bază ale limbajului de programare;
- CS4. Implementarea tipurilor structurate de date;
- CS5. Realizarea aplicațiilor pentru rezolvarea diverselor tipuri de probleme.

IV. Administrarea modului

Semestrul	Numărul de ore				Modalitatea de evaluare	Numărul de credite
	Total	Contact direct		Lucrul individual		
		Prelegeri	Practică/ Seminar			
V	90	30	30	30	examen	3

V. Unitățile de învățare

Unități de competență	Unități de conținut	Abilități
1. Elementele și vocabularul limbajului		
UC1. Identificarea etapelor de realizare a unui program și a elementelor.	1. Interfața mediului de programare. 2. Etape de realizare a unui program. 3. Structura a unui program. 4. Identificatori. 5. Constante. 6. Variabile. 7. Tipuri simple de date.	A1. Reprezentarea schematică a interfeței. A2. Selectarea meniurilor mediului de programare. A3. Recunoașterea opțiunilor ale meniurilor. A4. Utilizarea opțiunilor meniurilor interfeței. A5. Stabilirea etapelor de realizare a unui program. A6. Descrierea structurii unui program. A7. Definirea corectă a noțiunii de identificator. A8. Definirea corectă a noțiunii de constantă. A9. Definirea corectă a noțiunii de variabilă. A10. Identificarea tipurilor simple de date.
UC2. Utilizarea operatorilor și instrucțiunilor.	8. Clase de operatori. 9. Expresii. 10. Operatori specifici limbajului C. 11. Operații aritmetice. 12. Instrucțiuni.	A11. Definirea corectă a noțiunii de operator. A12. Identificarea claselor de operatori. A13. Definirea corectă a noțiunii de expresie. A14. Identificarea componentelor unei expresii. A15. Descrierea corectă a expresiilor. A16. Identificarea operatorilor specifici limbajului. A17. Definirea corectă a operațiilor aritmetice.

Unități de competență	Unități de conținut	Abilități
		A18. Realizarea expresiilor cu operații aritmetice. A19. Utilizarea operatorilor. A20. Definirea corectă a instrucțiunilor. A21. Aplicarea instrucțiunilor.
2. Funcții		
UC3. Utilizarea funcțiilor	13. Tipuri de funcții. 14. Funcții definite de utilizator. 15. Funcții de bibliotecă. 16. Funcții matematice. 17. Funcții de conversie. 18. Funcții de intrare-ieșire.	A22. Definirea corectă a funcției. A23. Clasificarea funcțiilor. A24. Descrierea noțiunii de funcție definită de utilizator. A25. Distingerea funcțiilor de intrare/ieșire. A26. Identificarea funcțiilor matematice. A27. Descrierea funcțiilor de conversie. A28. Utilizarea funcțiilor.
3. Structuri de date		
UC4. Aplicarea structurilor de date.	19. Noțiune de structură. 20. Structuri și tipuri definite de utilizator. 21. Operații cu tipul mulțime. 22. Tipul enumerat. 23. Tablouri de structuri.	A29. Declararea structurii. A30. Selectarea elementelor structurii. A31. Elaborarea tipurilor definite de utilizator. A32. Asignarea de nume pentru tipuri de date. A33. Definirea mulțimii. A34. Identificarea tipului enumerat. A35. Aplicarea tablourilor de structuri.

Unități de competență	Unități de conținut	Abilități
4. Pointeri		
UC5. Utilizarea pointerilor.	24. Alocarea dinamică a memoriei. 25. Referințe. 26. Legătura dintre pointeri și tablouri. 27. Pointeri spre funcții.	A36. Declararea de pointeri. A37. Aplicarea referințelor. A38. Realizarea operațiilor de pointeri. A39. Distingerea legăturii dintre pointeri și tablouri. A40. Aplicarea operatorilor și funcțiilor de alocare și eliberare a memoriei. A41. Definirea pointerilor la funcții.
5. Liste		
UC6. Utilizarea listelor.	28. Stiva. 29. Lista simplu înlănțuită. 30. Lista circulară simplu înlănțuită. 31. Accesul la un element al listei circulare simplu înlănțuite. 32. Lista dublu înlănțuită. 33. Accesul la un element al unei liste dublu.	A42. Definirea corectă a stivei. A43. Identificarea listei simplu înlănțuită. A44. Distingerea elementelor listei circulare. A45. Crearea unei liste circulare simplu înlănțuite. A46. Accesarea elementelor listei circulare simplu înlănțuite. A47. Identificarea listei simplu înlănțuită. A48. Distingerea elementelor listei dublu înlănțuită. A49. Crearea unei liste dublu înlănțuite. A50. Accesarea elementelor listei dublu.
UC7. Utilizarea fișierelor.	34. Tipuri de fișiere. 35. Operații cu fișiere. 36. Nivelul superior de prelucrare a fișierelor.	A51. Identificarea tipului de fișier. A52. Realizarea operațiilor cu fișiere. A53. Distingerea nivelelor de prelucrare a fișierelor.

Unități de competență	Unități de conținut	Abilități
6. Clase		
UC8. Utilizarea claselor.	37. Clase in. 38. Proprietățile claselor. 39. Tipuri de membri a unei clase. 40. Obiecte.	A54. Definirea unei clase. A55. Descrierea formei generale de declarație a unei clase. A56. Identificarea proprietăților unei clase. A57. Distingerea membrilor unei clase. A58. Identificarea tipului de membru al clasei. A59. Crearea si distrugerea obiectelor.

VI. Repartizarea orientativă a orelor pe unități de învățare

Nr. crt.	Unități de învățare	Numărul de ore			
		Total	Contact direct		Lucrul individual
			Prelegeri	Practică/Seminar	
1.	Elementele și vocabularul limbajului.	8	2	2	4
2.	Funcții.	16	4	6	6
3.	Structuri de date.	18	6	8	4
4.	Pointeri.	14	6	4	4
5.	Liste.	16	4	6	6
6.	Clase.	18	8	4	6
	Total	90	30	30	30

VII. Studiu individual ghidat de profesor

Materii pentru studiul individual	Produse de elaborat	Modalități de evaluare	Termeni de realizare
1. Elementele și vocabularul limbajului			
Structura a unui program.	Studiu de caz	Comunicare Demonstrare	Săptămâna 3
2. Funcții			
Tipuri de funcții.	Proiect individual	Prezentarea proiectului	Săptămâna 5
3. Structuri de date			
Structuri și tipuri definite de utilizator	Proiect în grup.	Prezentarea proiectului	Săptămâna 7
4. Pointeri			
Alocarea dinamică a memoriei.	Proiect în grup.	Prezentarea proiectului	Săptămâna 9
5. Liste			
Lista simplu și dublu înlănțuită. Operații cu fișiere.	Studiu de caz	Comunicare Demonstrare	Săptămâna 11

Materii pentru studiul individual	Produse de elaborat	Modalități de evaluare	Termeni de realizare
6. Clase			
Tipuri de membri a unei clase.	Studiu de caz	Comunicare Demonstrare	Săptămâna 13

VIII. Lucrările de laborator recomandate

1. Descrierea componentelor interfeței mediului de programare.
2. Studiarea opțiunilor meniurilor mediului de programare..
3. Prelucrarea tablourilor și șirurilor de caractere.
4. Rezolvarea problemelor cu funcții de utilizator.
5. Rezolvarea problemelor cu funcții de bibliotecă.
6. Rezolvarea problemelor cu funcții matematice.
7. Rezolvarea problemelor cu funcții de conversie.
8. Rezolvarea problemelor cu funcții de intrare-ieșire.
9. Rezolvarea problemelor cu pointeri.
10. Realizarea legăturii dintre pointeri și tablouri.
11. Prelucrarea listei simplu înlănțuită.
12. Prelucrarea listei circulare simplu înlănțuită.
13. Prelucrarea listei dublu înlănțuită.
14. Rezolvarea de probleme cu fișiere.
15. Rezolvarea de probleme cu clase.

IX. Sugestii metodologice

Curriculumul modular "Limbe de programare" este un curs de inițiere a elevilor în conceptele de bază de elaborarea aplicațiilor într-un limbaj de programare.

Activitățile de predare-învățare se recomandă a fi desfășurate într-o sală echipată cu calculatoare, tablă interactivă, conexiune la Internet – pentru îmbunătățirea instruirii interactive. La lecțiile practice și de laborator este necesar să existe un număr de stații de lucru egal cu numărul elevilor din clasă.

Volumul și varietatea conținutului materiei de studiu impun utilizarea unei varietăți de metode, mijloace și tehnici în procesul de instruire. Pentru o mai bună orientare metodele de învățământ utilizate în procesul didactic pot fi clasificate astfel:

- Jocuri de rol și simulare.
- Studii și cercetare.

- Tehnici pentru dezvoltarea gândirii critice.
- Exerciții și probleme.

Metoda **studiul de caz** valorifică o situație reală care se analizează și se rezolvă. În procesul de planificare a unei aplicații web apar situații când este necesar de a prezenta elevului aplicații deja elaborate. Avantajul metodei, constă în faptul că fiecare dintre elev își va aduce aportul la analiza aplicației web. În utilizarea acestei metode se conturează câteva etape:

- 1) Selectarea și prezentarea cazului.
- 2) Prelucrarea și conceptualizarea.
- 3) Structurarea finală a studiului.

Această metodă se recomandă în cazul studierii unităților de învățare: *Etapele de realizare a unui program; Structura a unui program; Interfața mediului de programare; Lista simplă dublu înlanțuită. Operații cu fișiere; Tipuri de membri a unei clase.*

Instruirea prin proiecte reprezintă o modalitate de instruire/autoinstruire grație căreia elevii, dar mai ales elevii efectuează o cercetare orientată spre obiective practice și finalizată într-un produs ce poate fi o schiță, o prezentare, o culegere tematică-informațională despre părți componente, un album cu imagini etc.

Această metodă se recomandă în cazul studierii unităților de învățare: *Tipuri de funcții; Structuri și tipuri definite de utilizator; Alocarea dinamică a memoriei.*

Procesul de studiu la Proiectarea asistată de calculator va deveni unul eficient dacă va fi însoțit de **Portofoliul activităților individuale** al elevului. Se recomandă, ca de-a lungul cursului, elevii să mențină un portofoliu structurat în două secțiuni. În partea I elevii colectează și analizează informații, știri și inovații în domeniul tehnologiilor web. În partea a II elevii vor reflecta asupra învățării prin realizarea activităților practice și individuale propuse la fiecare unitate de învățare. Această secțiune a portofoliului ar trebui să reflecte o colecție de experiențe și activități pe care elevii le-au realizat în cadrul orelor de activitate individuală pe tot parcursul anului. Aceasta este o oportunitate pentru profesor de a personaliza procesul de studiu și de a colecta produsele activității individuale ale elevului, deoarece acestea reflectă interesele, abilitățile și experiențele specifice de învățare ale fiecărui elev. Produsele, care vor fi prezentate în această secțiune pot fi foarte variate – rezolvări de probleme, analize, studii de caz, sarcini realizate pe teren/comunitate, interviuri, cercetări de piață ș.a.

Metodele recomandate pentru fiecare din unitățile de învățare sunt prezentate în tabelul de mai jos.

Nr. crt.	Unitatea de învățare	Metodele de învățare recomandate		
		Prelegeri	Practică	Individual
1.	Elemente și vocabularul limbajului.	Expunerea. Conversația euristică. Demonstrația.	Demonstrația. Observația. Problematizarea.	Realizarea studiului de caz. Activități creative.

Nr. crt.	Unitatea de învățare	Metodele de învățare recomandate		
		Prelegeri	Practică	Individual
2.	Functii.	Exercițiu. Expunerea. Demonstrația. Joc de rol.	Asaltul de idei. Descoperirea. Problematizarea. Elaborarea de proiecte.	Modelarea. Elaborarea de proiecte. Activități creative.
3.	Structuri de date.	Studiul de caz. Conversația euristică. Demonstrația. Joc de rol.	Modelarea. Simularea. Problematizarea.	Modelarea. Elaborarea de proiecte. Activități creative.
4.	Pointeri.	Studiul de caz. Descoperirea. Asaltul de idei. Exercițiu. Demonstrația. Joc de rol.	Problematizarea. Metodele de simulare.	Elaborarea de proiecte. Modelarea. Activități creative.
5.	Liste.	Studiul de caz. Exercițiu. Expunerea. Conversația euristică. Demonstrația. Joc de rol.	Problematizarea. Simularea. Demonstrația. Observația. Studiu de caz.	Studiu de caz. Modelarea. Activități creative.
6.	Clase.	Studiul de caz. Exercițiu. Expunerea. Conversația euristică. Demonstrația. Joc de rol.	Problematizarea. Simularea. Demonstrația. Observația. Studiu de caz.	Studiu de caz. Modelarea. Activități creative.

X. Sugestii de evaluare a competențelor profesionale

Activitățile de evaluare vor fi orientate spre motivarea elevilor și obținerea unui feedback continuu, fapt ce va permite corectarea operativă a procesului de învățare, stimularea autoevaluării și a evaluării reciproce, evidențierea succeselor, implementarea evaluării selective sau individuale.

Axarea procesului de învățare-predare-evaluare pe competențe presupune efectuarea evaluării pe parcursul întregului proces de instruire. Evaluarea continuă va fi structurată în

evaluări formative și evaluări sumative (finale) ce țin de interpretarea creativă a informațiilor și de capacitatea de a rezolva situațiile de problemă.

Pentru a eficientiza procesele de evaluare, înainte de a demara evaluările, cadrul didactic va aduce la cunoștința elevilor tematica lucrărilor, modul de evaluare (bareme/grile/criterii de notare) și condițiile de realizare a fiecărei evaluări.

Produsele recomandate pentru evaluarea nivelului de dezvoltare a competențelor cognitive sunt prezentate mai jos:

1. Exercițiu rezolvat:
 - Înțelegerea enunțului exercițiului.
 - Corectitudinea formulării ipotezelor.
 - Corectitudinea raționamentelor.
 - Corectitudinea testării ipotezelor.
 - Corectitudinea strategiei rezolutive.
 - Corectitudinea rezultatelor.
 - Modul de prezentare a rezultatelor.
 - Modul de interpretare a rezultatelor.
2. Problemă rezolvată:
 - Înțelegerea problemei.
 - Documentarea în vederea identificării informațiilor necesare în rezolvarea problemei.
 - Formularea și testarea ipotezelor.
 - Stabilirea strategiei rezolutive.
 - Prezentarea și interpretarea rezultatelor.
3. Proiect elaborat:
 - Validitatea proiectului - gradul în care acesta acoperă unitar și coerent, logic și argumentat tema propusă.
 - Completitudinea proiectului - felul în care au fost evidențiate conexiunile și perspectivele interdisciplinare ale temei, competențele și abilitățile de ordin teoretic și practic și maniera în care acestea servesc conținutului științific.
 - Elaborarea și structura proiectului - acuratețea, rigoarea și coerența demersului științific, logica și argumentarea ideilor, corectitudinea concluziilor.
 - Calitatea materialului folosit în realizarea proiectului, bogăția și varietatea surselor de informare, relevanța și actualitatea acestora, semnificația datelor colectate s.a.
 - Creativitatea - gradul de noutate pe care-l aduce proiectul în abordarea temei sau în soluționarea problemei.
4. Rezumat oral:
 - Expune tematică a lucrării în cauză.
 - Utilizează formulări proprii, fără a distorsiona mesajul lucrării supuse rezumării.
 - Expunerea orală este concisă și structurată logic.
 - Folosește un limbaj bogat, adecvat tematicii lucrării în cauză.
 - Respectarea coeficientului de reducere a textului: 1/3 din textul inițial.
5. Rezumat scris:
 - Expune tematica lucrării în cauză.

- Utilizează formulări proprii, fără a distorsiona mesajul lucrării supuse rezumării.
 - Textul rezumatului este concis și structurat logic.
 - Folosește un limbaj bogat, adecvat tematicii lucrării în cauză.
 - Fidelitatea: înțelegerea esențialului și reproducerea lui, nu trebuie să existe contrasens.
 - Coerența: rezumatul are o unitate și un sens evident, lizibil pentru cei care nu cunosc textul sursă.
 - Progresia logică: înlănțuirea ideilor, prezentarea argumentelor sunt clare și evidente.
 - Angajamentul autorului, aptitudine critică corect evaluată și transpusă.
 - Respectarea modalităților de enunțare a textului sursă: rezumatul este o oglindă micșorată dar fidelă textului sursă.
 - Muncă pertinentă de reformulare: rezumatul nu este un colaj de citate.
 - Respectarea coeficientului de reducere a textului: 1/4 din textul inițial.
 - Stăpânirea normelor sintactice la nivel de prezentare logică a ideilor, frazelor, paragrafelor textului.
 - Text formatat citet, lizibil. plasarea clară în pagină.
6. Studiu de caz:
- Corectitudinea interpretării studiului de caz propus.
 - Calitatea soluțiilor, ipotezelor propuse, argumentarea acestora.
 - Corespunderea soluțiilor, ipotezelor propuse pentru rezolvarea adecvată a cazului analizat.
 - Corectitudinea lingvistică a formulărilor.
 - Utilizarea adecvată a terminologiei în cauză.
 - Rezolvarea corectă a problemei, asociate studiului analizat de caz.
 - Punerea în evidență a subiectului, problematicii și formularea.
 - Logica sumarului.
 - Referință la programe aplicate.
 - Completitudinea informației și coerența între subiect și documentele studiate.
 - Noutatea și valoarea științifică a informației.
 - Exactitatea rezultatelor și rigoarea probelor.
 - Capacitatea de analiză și de sinteză a documentelor, adaptarea conținutului.
 - Originalitatea studiului, a formulării și a realizării.
 - Personalizarea (să nu fie lucruri copiate).
 - Aprecierea critică, judecată personală a elevului.
 - Corectitudinea interpretării studiului de caz propus.
 - Calitatea soluțiilor, ipotezelor propuse, argumentarea acestora.
 - Corespunderea soluțiilor, ipotezelor propuse pentru rezolvarea adecvată a cazului analizat.
 - Corectitudinea lingvistică a formulărilor.
 - Rezolvarea corectă a problemei, asociate studiului analizat de caz.
7. Item electronic rezolvat:
- Corectitudinea interpretării itemului propus spre rezolvare.
 - Corespunderea rezolvării propuse de condițiile indicate în item.
 - Corectitudinea metodei utilizate de rezolvare.
 - Corectitudinea selectării răspunsului (pentru itemi cu alegere duală).
 - Integritatea și corectitudinea setului de selecții (pentru itemi cu alegere multiplă).
 - Stabilirea corectă a perechilor corelate (pentru itemii tip asociere).

- Corespunderea răspunsului setului prestabilit de valori.
 - Corespunderea răspunsului cerințelor din enunțul itemului (în cazul itemilor cu răspuns deschis).
 - Localizarea corectă a componentelor în schemă.
 - Calitatea grafică a prezentării răspunsului.
8. Test de laborator realizat:
- Corectitudinea realizării aplicației.
 - Acuratețea realizării calculelor.
 - Aplicarea corectă a tehnicilor de programare.
 - Amplasarea clară pe ecran a datelor de intrare și celor de ieșire.

În calitate de produse pentru evaluarea competențelor profesionale se vor folosi:

- schema interfeței mediului de programare;
- texte – sursă ale aplicației realizată;
- screenshot-ul cu rezultatul prelucrării tablourilor și șirurilor de caractere;
- screenshot-ul cu rezultatul rezolvării problemelor cu funcții de utilizator.
- screenshot-ul cu rezultatul rezolvării problemelor cu funcții de bibliotecă.
- screenshot-ul cu rezultatul rezolvării problemelor cu funcții matematice.
- screenshot-ul cu rezultatul rezolvării problemelor cu funcții de conversie.
- screenshot-ul cu rezultatul rezolvării problemelor cu funcții de intrare-ieșire.
- screenshot-ul cu rezultatul rezolvării problemelor cu pointeri.
- screenshot-ul cu rezultatul prelucrării listei simplu înlănțuită.
- screenshot-ul cu rezultatul prelucrării listei circulare simplu înlănțuită.
- screenshot-ul cu rezultatul prelucrării listei dublu înlănțuită.
- screenshot-ul cu rezultatul rezolvării de probleme cu fișiere.
- screenshot-ul cu rezultatul rezolvării de probleme cu clase.

Criteriile pentru evaluarea sumativă a competențelor profesionale sunt prezentate mai jos:

1. Utilizarea elementelor de interfață ale mediului de programare integrat:
 - Corectitudinea utilizării mediului de programare integrat.
 - Corectitudinea setărilor implicite.
 - Productivitatea muncii.
2. Identificarea etapelor de realizare a unui program:
 - Respectarea succesiunii etapelor;
 - Corectitudinea realizării acțiunilor la fiecare etapă;
 - Relevanța cerințelor față de rezultatele problemei;
 - Completitudinea cerințelor față de rezultatele problemei.
3. Aplicarea elementelor de bază ale limbajului de programare:
 - Corectitudinea elementelor;
 - Gradul de redare a esenței elementelor;
 - Creativitatea și originalitatea.
 - Originalitatea aplicării elementelor în corelare cu redarea cerințelor și regulile de programare;
 - Impactul vizual și comunicativ a rezultatelor obținute.

4. Implementarea tipurilor structurate de date:
 - Corectitudinea datelor în baza cărora se implementează tipurile structurate de date.
 - Relevanța cerințelor față de structurile de date.
 - Completitudinea cerințelor față de structurile de date.
5. Realizarea aplicațiilor pentru rezolvarea diverselor tipuri de probleme:
 - Corectitudinea perceperii esenței problemei.
 - Gradul de exactitate a redării rezultatelor problemei rezolvate.
 - Oportunitatea datelor de intrare și ieșire.
 - Relevanța datelor utilizate.
 - Creativitatea și originalitatea.
 - Impactul vizual și comunicativ.

XI. Resursele necesare pentru desfășurarea procesului de studii

Cerințe față de sălile de curs	
Pentru orele teoretice	Tablă interactivă sau proiector
Pentru orele de laborator	Laborator de informatică care asigură fiecărui elev un calculator.
Cerințe tehnice	
Parametri tehnici minimi ale calculatorului	Procesor: 2 GHz Memorie operativă: 4 GB Unitate de stocare: 500 GB Rețea: Ethernet, 100 Mbps
Software	Sistem de Operare Microsoft Windows. Mediul de programare.

XII. Resursele didactice recomandate elevilor

Nr. crt.	Denumirea resursei	Locul în care poate fi consultată/ accesată/ procurată resursa	Numărul de exemplare disponibile
1.	Emanuela Cerchez, Marinel Serban Programarea în limbajul C/C++ pentru liceu – vol. I, II, III, IV., POLIROM		
2.	Nicolae Constantinescu, Programare Procedurală, Editura Universitaria, 2009	Biblioteca	
3.	Doina Logofătu, Algoritmi fundamentali în C++. Aplicații, Ed. 1, Editura Polirom, Iași, 2007	Biblioteca	

Nr. crt.	Denumirea resursei	Locul în care poate fi consultată/ accesată/ procurată resursa	Numărul de exemplare disponibile
4.	B. Kernigham, D. Ritchie - Limbajul de programare C, editia a 2-a, Ed. Teora, 2003.	Biblioteca	
5.	M. Cosulschi, M. Gabroveau - Algoritmi - o abordare pragmatica, editia a 2-a, Ed. Universitaria, 2003.	Biblioteca	
6.	С. Седжвик, Язык программирования Си++. М.: Мир, 2001.	Biblioteca	
7.	C. Botez, M. Gospodaru, Limbaje de programare. Introducere în limbajul C. Iași, 1998.	Biblioteca	–
8.	N. Barkakati, Borland C++. Ghidul programatorului. București: Teora, 1997.	Biblioteca	–
9.	C. Namir, Shammass, Curs rapid de Borland C++ 4. București: Teora, 1996 (trad.engl.).	Biblioteca	–
10.	http://www.cplusplus.com/	Internet	
11.	http://infoscience.3x.ro/c++.html	Internet	
12.	http://www.studytonight.com/cpp/introduction-to-cpp.php	Internet	
13.	http://www.aut.upt.ro/~dberian/Labs/POO.pdf	Internet	
14.	http://www.scribub.com/stiinta/informatica/c/	Internet	