



Ministerul Educației și Cercetării al Republicii Moldova

Centrul de Excelență în Energetică și Electronică

Aprob
Directorul Centrului de Excelență
în Energetică și Electronică

Mariana BARLADEAN
16 ianuarie 2023

Curriculum disciplinar

F.04.O.014 Limbaje de programare

Specialitatea: 61110 – Calculatoare

Calificarea: Tehnician pentru suportul tehnic al calculatoarelor

Curriculumul a fost elaborat în baza ordinului Ministerului Educației și Cercetării al Republicii Moldova, numărul 83, din data de 14.02.2022, cu privire la aprobarea listei instituțiilor de învățământ desemnate responsabile pentru elaborarea planurilor de învățământ la programele de formare profesională postsecundară și postsecundar nonterțială



Autori:

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

Josan Liudmila, grad didactic întâi, Colegiul Politehnic mun. Bălți.

Aprobat de:

Consiliul metodic-științific al Centrului de Excelență în Energetică și Electronică.

Director adjunct pentru instruire

ZVEZDENCO Gheorgii

16 ianuarie 2023

Recenzenți:

1. GAMA COMPUTER SRL adresa: str. Meșterul Manole, 12/2, Chișinău,
Director: Minchevici Sergiu.
2. VIC-COM INFO SRL, adresa: str. Albișoara 68/3 of. 72, MD, mun. Chișinău, Director:
Tabuci Victor.

Adresa Curriculumului în Internet:

<https://ceee.md/programe-de-formare-profesionala>

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.....	7
VII. Studiu individual ghidat de profesor	7
VIII. Lucrările de laborator recomandate	8
IX. Sugestii metodologice.....	8
X. Sugestii de evaluare a competențelor profesionale.....	10
XI. Resursele necesare pentru desfășurarea procesului de studii.....	13
XII. Resursele didactice recomandate elevilor.....	14

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.

Disciplina respectivă este structurată în șase unități de învățare și este destinată 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.

II. Motivația, utilitatea modului pentru dezvoltarea profesională

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

La finalizarea disciplinei, 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 disciplinei

CS1. Operarea cu fundamente ale informaticii specifice programării calculatoarelor într-un limbaj de programare (limbajul C/C++).

CS2. Însușirea elementelor fundamentale privind paradigmele de programare structurată, procedurală și modulară, respectiv asimilarea fundamentelor limbajului de programare.

CS3. Implementarea algoritmilor de rezolvare a problemelor de calcul numeric din diferite domenii ale științei și tehnicii.

CS4. Formarea abilităților de comunicare interdisciplinară, organizare și management al lucrului în echipă.

IV. Administrarea disciplinei

Semestrul	Numărul de ore			Modalitatea de evaluare	Numărul de credite	
	Total	Contact direct				Lucrul individual
		Prelegeri	Practică/ Seminar			
IV	120	30	30	60	examen	4

V. Unitățile de învățare

Unități de competență	Unități de conținut
1. Bazele programării în C/C++	
1. Editarea fragmentelor de program: - scrierea simbolurilor speciale în limbajul de programare; - identificarea cuvintelor cheie ale limbajului de programare; - declararea variabilelor/constantelor conform tipului de dată solicitat; - selectarea operatorilor în dependență de specificul tipului de dată utilizat; - identificarea operatorilor și operanzilor; - utilizarea operatorilor și operanzilor la prelucrarea datelor.; - identificarea părților componente ale unui program.	1. Programarea și limbaje de programare. 2. Elemente de bază ale limbajului C/C++ . - Tipuri de date - Modificatorii de tip - Constante - Variabile 2. Operatorii limbajului C/C++. - Operatori aritmetici - Operatori relationali - Operatori de egalitate 3. Preprocesare. Structura unui program C/C++.
2. Instrucțiuni de control	
2. Utilizarea instrucțiunilor în cadrul aplicațiilor de consolă: - translarea algoritmilor cu decizie în limbajul de programare; - elaborarea algoritmilor de decizie conform specificațiilor propuse; - implementarea algoritmilor de decizie în limbajul de programare; - identificarea părților componente ale unei instrucțiuni ciclice; - elaborarea algoritmilor ciclici cu un număr cunoscut de pași; - elaborarea algoritmilor ciclici cu un număr necunoscut de pași; - elaborarea algoritmilor ciclici cu întrerupere forțată; - translarea algoritmilor ciclici în limbajul de programare; - implementarea algoritmilor ciclici în limbajul de programare.	4. Instrucțiuni de decizie. 5. Instrucțiuni de decizie multiplă. 6. Instrucțiunea ciclică for. 7. Instrucțiunea ciclică while. Instrucțiunea do_while.
3. Pointeri	
3. Utilizarea pointerilor: - declararea de pointeri; - aplicarea referințelor; - realizarea operațiilor de pointeri; - aplicarea operatorilor și funcțiilor de alocare și eliberare a memoriei.	8. Pointeri. Declararea pointerilor. Alocarea dinamică a memoriei. 9. Operații asupra pointerilor.
4. Funcții	
4. Utilizarea funcțiilor: - definirea corectă a funcției;	10. Declararea funcției. Apelul funcției. Prototipul funcției.

Unități de competență	Unități de conținut
- clasificarea funcțiilor; - descrierea noțiunii de funcție definită de utilizator; - identificarea funcțiilor matematice; - descrierea funcțiilor de conversie; - utilizarea funcțiilor.	11. Parametri formali și actuali. Variabile locale și variabile globale. Variabile statice și variabile automate. 12. Funcții matematice.
5. Tipuri de date structurate	
5. Prelucrarea tablourilor în cadrul aplicațiilor de consolă: - declararea tipurilor de date tablou; - inițializarea elementelor unui tablou unidimensional/ bidimensional; - citirea/afișarea elementelor tablourilor unidimensionale/ bidimensionale; - prelucrarea elementară a tablourilor unidimensionale/ bidimensionale; - declararea șirurilor de caractere; - utilizarea șirurilor de caractere în cadrul aplicațiilor; - utilizarea funcțiilor standard la prelucrarea șirurilor de caractere; - declararea structurii; - selectarea elementelor structurii; - prelucrarea tipurilor definite de utilizator.	13.Tablouri unidimensionale. 14.Tablouri bidimensionale. 15.Șiruri de caractere. Funcții specifice șirurilor de caractere 16.Noțiune de structură. Structuri și tipuri definite de utilizator.
6. Fișiere	
6. Utilizarea fișierelor: - Identificarea tipului de fișier; - realizarea operațiilor cu fișiere; - distingerea nivelelor de prelucrare a fișierelor; - prelucrarea fișierelor în limbajul C/C++.	17.Declarare. Operații cu fișiere. - Prelucrarea fișierelor la nivel inferior - Prelucrarea fișierelor la nivel superior 18.Fișiere în C++.

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.	Bazele programării în C/C++	12	4	2	6
2.	Instrucțiuni de control	24	6	6	12
3.	Pointeri	16	4	4	8
4.	Funcții	20	6	4	10
5.	Tipuri de date structurate	28	6	8	14
6.	Fișiere	20	4	6	10
	Total	120	30	30	60

VII. Studiu individual ghidat de profesor

Materii pentru studiul individual	Produse de elaborat	Modalități de evaluare	Termeni de realizare
1. Bazele programării în C/C++			
Programarea și limbaje de programare. Elemente de bază ale limbajului C/C++ . Preprocesare. Structura unui program C/C++.	Studiu de caz	Comunicare Demonstrare	Săptămâna 3
2. Instrucțiuni de control			
Aplicarea instrucțiunilor de control în rezolvarea sarcinilor concrete.	Proiect individual	Prezentareaproiectului	Săptămâna 5
3. Pointeri			
Alocarea dinamică a memoriei.	Studiu de caz	Prezentareaproiectului	Săptămâna 7
4. Funcții			
Funcții definite de utilizator	Proiect în grup.	Prezentareaproiectului	Săptămâna 9
5. Tipuri de date structurate			
Aplicarea tipurilor de date structurate în practică.	Proiect individual	Comunicare Demonstrare	Săptămâna 11
6. Fișiere			
Prelucrarea fișierelor în limbajul C++	Studiu de caz	Comunicare Demonstrare	Săptămâna 13

VIII. Lucrările de laborator recomandate

1. Limbaj de programare C/C++. Crearea de programe în C/C++. Algoritmi liniari.
2. Instrucțiuni condiționate.
3. Instrucțiuni ciclice.
4. Pointeri.
5. Funcții definite de utilizator.
6. Recursivitate.
7. Tablouri unidimensionale.
8. Tablouri bidimensionale.
9. Prelucrarea șirurilor de caractere.
10. Structuri.
11. Fișiere.

IX. Sugestii metodologice

Curriculumul modular "Limbaje 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 unui program; Interfața mediului de programare; Lista simplă dublu înlănțuită. 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.	Bazele programării în C/C++	Expunerea Conversația euristică Demonstrația	Demonstrația. Observația. Problematizarea.	Realizarea studiului de caz. Activități creative.
2.	Instrucțiuni de control	Exercițiu. Expunerea. Demonstrația. Joc de rol.	Asaltul de idei. Descoperirea. Problematizarea. Elaborarea de proiecte.	Modelarea. Elaborarea de proiecte. Activități creative.
3.	Pointeri	Studiul de caz. Conversația euristică. Demonstrația. Joc de rol.	Modelarea. Simularea. Problematizarea.	Modelarea. Elaborarea de proiecte. Activități creative.
4.	Funcții	Studiul de caz. Descoperirea. Asaltul de idei. Exercițiu. Demonstrația. Joc de rol.	Problematizarea. Metodele de simulare.	Elaborarea de proiecte. Modelarea. Activități creative.
5.	Tipuri de date structurate	Studiul de caz. Exercițiu. Expunerea. Conversația euristică. Demonstrația. Joc de rol.	Problematizarea. Simularea. Demonstrația. Observația. Studiul de caz.	Studiul de caz. Modelarea. Activități creative.
6.	Fișiere	Studiul de caz. Exercițiu. Expunerea. Conversația euristică. Demonstrația. Joc de rol.	Problematizarea. Simularea. Demonstrația. Observația. Studiul de caz.	Studiul 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 ordinteoretic 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 ș.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 supuserezumării.
- 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 supuserezumării.
- 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 corect, 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 sumarulului.
- Referință la programe aplicate.
- Complectitudinea 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 in limbajul C/C++ pentru liceu – vol. I, II, III, IV., POLIROM	Biblioteca	2
2.	Nicolae Constantinescu, Programare Procedurala, Editura Universitaria, 2009	Biblioteca	1
3.	Doina Logofătu, Algoritmi fundamentali in C++. Aplicații, Ed. 1, Editura Polirom, Iași, 2007	Biblioteca	3
4.	B. Kernigham, D. Ritchie - Limbajul de programare C, editia a 2-a, Ed. Teora, 2003.	Biblioteca	1
5.	M. Cosulschi, M. Gabroveau - Algoritmi - o abordare pragmatica, editia a 2-a, Ed. Universitaria, 2003.	Biblioteca	1
6.	С. Седжвик, Язык программирования Си++. М.: Мир, 2001.	Biblioteca	1
7.	C. Botez, M. Gospodaru, Limbaje de programare. Introducere în limbajul C. Iași, 1998.	Biblioteca	3
8.	N. Barkakati, Borland C++. Ghidul programatorului. București: Teora, 1997.	Biblioteca	1
9.	C. Namir, Shammass, Curs rapid de Borland C++ 4. București: Teora, 1996 (trad.engl.).	Biblioteca	1
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	
14.	http://www.scribub.com/stiinta/informatica/c/	Internet	