Ministerul Educaţiei și Cercetării al Republicii Moldova
Centrul de Excelență în Informatică şi Tehnologii Informaționale

"Aprob"
Directorul Centrului de Excelență în Informatică şi Tehnologii Informaționale



_____ Vitalie Zavadschi

Curriculumul modular
F.02.O.012 Programarea procedurală

Specialitatea: **61310 Programarea și analiza produselor program**
Calificarea: **Asistent programator**

Chișinău 2022

Aprobat de:
Consiliul metodico-ştiinţific al Centrului de Excelenţă în Informatică şi Tehnologii Informaţionale, proces verbal Nr.3 din 31.10.2022

Director adjunct _____
Obadă Liuba

Autori:
Gîncu Silviu, doctor în pedagogie, grad didactic superior.
Pîrvan Evgheni, grad didactic superior, Colegiul „Iulia Hașdeu" din Cahul.

Actualizat în baza Planului de învățământ nr. SC-13/20, aprobat prin OMEC Nr.1310, din 23.11.2020:
Frunză Olga, grad didactic doi, Centrul de Excelență în Informatică și Tehnologii Informaționale

Recenzenți:
Aremesu Vitalie, director „EBS Integrator" SRL, adresa: str.Ion Inculeț 33, mun.Chișinău.
Nastasenco Veaceslav, director „Allied Testing-Moldova" SRL, adresa: str. Ismail 33/1, mun.Chișinău.

Adresa Curriculumului în Internet:
https://mecc.gov.md/ro/content/curriculum-invatamintul-profesional-tehnic-postsecundar.

Cuprins

I. Preliminarii....4

II. Motivația, utilitatea modulului pentru dezvoltarea profesională....4

III. Competențele profesionale specifice modulului....5

IV. Administrarea modulului....5

V. Unităţile de învățare....6

VI. Repartizarea orientativă a orelor pe unități de învățare....9

VII. Studiu individual ghidat de profesor....9

VIII. Lucrările practice recomandate....9

IX. Sugestii metodologice....10

X. Sugestii de evaluare a competențelor profesionale....12

XI. Resursele necesare pentru desfășurarea procesului de studii....14

XII. Resursele didactice recomandate elevilor....15

I. Preliminarii

Programarea procedurală este o paradigmă de programare bazată pe conceptul de apel de procedură (subprogram). Procedurile, numite şi rutine, subrutine, metode sau funcții conțin o serie de pași computaționali care trebuie parcurși. Orice procedură poate fi apelată oriunde în timpul execuției unui program inclusiv din interiorul ei (apel recursiv).

Statutul Curriculumului. Curriculumul modular "Programarea procedurală" este un document normativ şi obligatoriu pentru realizarea procesului de pregătire a tehnicienilor în învățământul profesional tehnic post secundar, care vor elabora diverse aplicații în conformitate cu sarcinile de lucru.

Funcțiile Curriculumului. Funcțiile de bază ale Curriculumului sunt:

- act normativ al procesului de predare, învățare, evaluare şi certificare în contextul unei pedagogii axate pe competente;
- reper pentru proiectarea didactică şi desfășurarea procesului educațional din perspectiva unei pedagogii axate pe competente;
- componentă de bază pentru elaborarea strategiei de evaluare şi certificare;
- orientare a procesului educațional spre formare de competente la elevi;
- componentă fundamentală pentru elaborarea manualelor tipărite, manualelor electronice, ghidurilor metodologice, testelor de evaluare.

Beneficiarii Curriculumului. Curriculumul este destinat:

- profesorilor din instituțiile de învăţământ profesional tehnic post secundar;
- autorilor de manuale şi ghiduri metodologice;
- elevilor care își fac studiile la specialitatea în cauză;
- membrilor comisiilor pentru examenele de calificare;
- membrilor comisiilor de identificare, evaluare şi recunoaștere a rezultatelor învățării, dobândite în contexte non-formale şi informale.

Scopul studierii acestui modul constă în formarea şi dezvoltarea competenței profesionale specifice de utilizarea a tipurilor de date structurate și a subprogramelor în mentenanța şi actualizarea produselor-program (softurilor) de sistem precum şi a altor aplicații. De asemenea, modulul contribuie la dezvoltarea competenței profesionale generale de respectare şi de promovare a normelor de drept informatic.

Modulul în cauză poate fi studiat după însușirea în mod obligatoriu a unității de curs:

F.01.O.010 Programarea structurată.

II. Motivația, utilitatea modulului pentru dezvoltarea profesională

Studierea acestui modul va contribui la formarea şi dezvoltarea de competențe profesionale ce corespund nivelului patru de calificare:

- cunoștințe faptice, principii, procese și concepte generale din domeniul elaborării produselor program;
- abilități cognitive și practice necesare pentru elaborarea aplicațiilor de consolă conform tematicilor incluse;

- asumarea responsabilității pentru mentenanța de aplicații.

Competențele formate şi dezvoltate în cadrul acestui modul vor fi necesare pentru studierea următoarelor module:

P.02.O.001 Practica de inițiere în specialitate
F.03.O.013 Programarea calculatorului
F.04.O.015 Asistență pentru programarea orientată pe obiecte
P.04.O.002 Practica de instruire
S.06.O.020 Asistență pentru programarea vizuală
S.06.O.021 Asistență pentru programare client-side a site-urilor Web
S.06.O.022 Asistență pentru programare server-side a site-urilor Web
S.08.A.024 Testarea și depănarea produselor program
S.08.O.025 Dezvoltarea aplicațiilor pentru dispozitive mobile

De asemenea, ele vor fi de un real folos în activitatea profesională a tehnicianului, în special, în ocupațiile legate de gestiunea produselor-program utilizate în companii.

III. Competențele profesionale specifice modulului

În cadrul modulului vor fi formate şi dezvoltate următoarele competențe profesionale specifice:

CS1. Prelucrarea tipurilor de date structurate în cadrul aplicațiilor de consolă;

CS2. Organizarea programelor la nivel de subprograme;

CS3. Elaborarea de programare pentru problemele întâlnite în activitatea profesională.

IV. Administrarea modulului

Semestrul	Numărul de ore				Forma de evaluare	Numărul de credite
	Total	Contact direct		Ore de studiu individual		
		Teorie	Laborator			
II	120	30	30	60	Examen	4

V. Unităţile de învăţare

Unităţi de competenţă	Unităţi de conţinut	Abilităţi
1. **Tipuri de date structurate**		
UC1. Utilizarea fișierelor în cadrul aplicațiilor de consolă	1. Tipul de date fișier. Operații specifice tipului de date fișier: – citire; – scriere; – funcții de prelucrare a fișierelor.	A1. Declararea variabilelor de tip fișier. A2. Deschiderea fișierelor conform specificațiilor propuse. A3. Închiderea fișierelor. A4. Verificarea sfârșitului de fișier. A5. Citirea datelor din fișiere. A6. Scrierea datelor în fișiere. A7. Utilizarea funcțiilor de prelucrare a fișierelor conform specificațiilor propuse.
UC2. Sortarea datelor structurate în cadrul aplicațiilor de consolă	2. Metode de sortare: – metoda de inserție; – metoda bulelor; – metoda de selecție; – sortare prin distribuire.	A8. Selectarea algoritmilor de sortare a datelor structurate. A9. Prezentarea situațiilor de aplicare a metodelor de sortare. A10. Detalierea algoritmilor de sortare a datelor structurate. A11. Translarea algoritmilor de sortare în limbajul de programare. A12. Implementarea algoritmilor de sortare în limbajul de programare.
UC3. Prelucrarea tipurilor de date de tip structură în cadrul aplicațiilor de consolă	3. Structuri. 4. Tablouri de structuri.	A13. Declararea structurilor de date conform specificațiilor propuse. A14. Declararea variabilelor de tip structură. A15. Declararea tablourilor de tip structură. A16. Inițializarea variabilelor de tip structură. A17. Accesarea câmpurilor unei structuri. A18. Citirea/afișarea datelor unei tip de date structură. A19. Afișarea structurilor. A20. Implementarea operațiilor asupra structurilor conform specificațiilor propuse. A21. Sortarea tablourilor de structuri. A22. Utilizarea tablourilor de structuri în cadrul aplicațiilor de consolă.

Unităţi de competenţă	Unităţi de conţinut	Abilităţi
UC4. Utilizarea pointerilor în cadrul aplicațiilor de consolă	5. Pointer. Variabile pointer. 6. Alocarea dinamică memoriei. Crearea dinamică a tipului de date tablou. 7. Relații dintre pointeri și diverse tipuri de date.	A23. Declararea unei variabile de tip pointer. A24. Utilizarea operatorului de adresare. A25. Utilizarea operatorului de diferențiere. A26. Setarea legăturilor pointer-tablou. A27. Accesarea datelor tablourilor unidimensionale/structurilor prin intermediul pointerului. A28. Aplicarea operațiilor asupra pointerilor conform specificațiilor propuse. A29. Alocarea dinamică a memoriei pentru tipurile de date numerice/caracter. A30. Alocarea dinamică a memoriei pentru tablouri. A31. Eliberarea memoriei.
2. **Subprograme**		
UC5. Utilizarea subprogramelor în cadrul aplicațiilor de consolă	8. Noțiune de subprograme. – subprograme cu tip; – subprograme fără tip. 9. Domenii de vizibilitate: – structura de bloc a programelor; – variabile globale şi variabile locale. 10. Efecte colaterale: – cauzele efectelor colaterale; – modul de evitare a efectelor colaterale.	A32. Utilizarea terminologiei specifice subprogramelor. A33. Declararea subprogramelor cu/fără tip. A34. Declararea anticipată a subprogramelor. A35. Apelarea subprogramelor cu/fără tip. A36. Elaborarea subprogramelor cu tip/fără tip. A37. Elaborarea subprogramelor cu parametri de tip valoare/adresă. A38. Prelucrarea datelor utilizate în procesul de studiu la nivel de subprograme. A39. Comunicarea între subprograme prin intermediul variabilelor globale/parametrilor. A40. Evitarea efectelor colaterale. A41. Utilizarea subprogramelor definite de către utilizator în cadrul programelor. A42. Prelucrarea datelor structurate la nivel de subprograme. A43. Elaborarea algoritmilor la nivel de subprograme. A44. Implementarea algoritmilor bazați pe subprograme în limbajul de programare.

Unităţi de competenţă	Unităţi de conţinut	Abilităţi
UC6. Utilizarea subprogramelor recursive în cadrul aplicațiilor de consolă	11. Subprograme recursive.	A45. Descrierea subprogramelor recursive. A46. Prezentarea situațiilor de aplicare a subprogramelor recursive. A47. Elaborarea subprogramelor recursive cu cel puțin doi parametri. A48. Elaborarea subprogramelor recursive indirecte. A49. Apelarea subprogramelor recursive. A50. Evitarea cazurilor de supraîncărcare a stivei. A51. Elaborarea algoritmilor recursivi pentru prelucrarea datelor utilizate în procesul de studiu. A52. Implementarea algoritmilor recursivi în limbajul de programare.
UC7. Organizarea aplicațiilor de consolă la nivel de module	12. Programarea modulară: – unit-uri (module); – supraîncărcare.	A53. Crearea unit-urilor. A54. Supraîncărcarea subprogramelor. A55. Utilizarea unit-urilor definite de către utilizator în cadrul aplicațiilor de consolă. A56. Elaborarea aplicațiilor de consolă bazate pe unit-uri. A57. Elaborarea proiectelor modulare conform specificațiilor propuse. A58. Gestiunea datelor tablourilor de tip structură la nivel de module.

VI. Repartizarea orientativă a orelor pe unități de învățare

Nr. crt.	Unități de învățare	Numărul de ore			
		Total	Contact direct		Ore de studiu individual
			Teorie	Laborator	
1.	Tipuri de date structurate	64	16	16	32
2.	Subprograme	56	14	14	28
	Total	**120**	**30**	**30**	**60**

VII. Studiu individual ghidat de profesor

Materii pentru studiul individual	Produse de elaborat	Forma de evaluare	Termeni de realizare
1. Tipuri de date structurate.			
Structuri. Metode de sortare.	Portofoliu: Set de aplicații de consolă cu utilizarea tipului de date fișier	Prezentarea portofoliului. Demonstrarea pe calculator.	Săptămâna 4-5
2. Subprograme			
Pointeri. Elaborarea de subprograme. Subprograme recursive	Portofoliu: Set de aplicații de consolă pentru prelucrarea structurilor.	Prezentarea portofoliului. Demonstrarea pe calculator.	Săptămâna 13-14

VIII. Lucrările practice recomandate

Lucrările practice vor fi efectuate în formă de lucrări de laborator. Tematica lucrărilor de laborator va include următoarele conținuturi:

1. Citirea şi scrierea datelor din/în fișiere.
2. Sortarea datelor.
3. Procesarea structurilor.
4. Pointeri.
5. Subprograme cu tip.
6. Subprograme fără tip.
7. Subprograme recursive.
8. Module/Unități de program.

IX. Sugestii metodologice

Elementul de bază al Curriculumului sunt competențele ce trebuie formate şi dezvoltate în procesul de instruire profesională.

Acestea vor fi formate prin organizarea eficientă a procesului de instruire. Pentru aceasta sunt necesare două condiții:

1. Organizarea activităților. Pentru buna organizare a procesului didactic ambii participanți necesită de a-şi organiza activitățile. De modul cum sunt organizate acestea depinde în mare măsură nivelul de formare a competențelor. În această ordine de idei, în procesul de organizare a activităților se vor asigura:

- condiții optime pentru buna colaborare dintre elev și profesor;
- un set de procese care duc la îmbunătățirea relațiilor dintre părți;
- un nivel de implicare a părților acționând în baza unor reguli şi acțiuni prestabilite.

2. Selectarea adecvată a metodelor de instruire. Se recomandă utilizarea metodelor de instruire precum:

Instruirea la distanță. Este o formă alternativă de învățământ în cadrul căreia se asigură continuarea procesului educațional în condițiile de carantină, prin intermediul diverselor instrumente de comunicare la distanță, cum ar fi: e-mail, chat, forum, blog. În calitate de resurse de învățare vor servi manualul și materialele didactice tradiționale care pot fi extinse prin resurse plasate online, iar în calitate de platforme Web necesare pentru desfășurarea instruirii la distanță se pot utiliza: Jitsi, WebEx, Skype, Moodle, Google Drive, Cisco Networking Academy, Google Classroom etc.

Simularea şi modelarea. Simularea este utilizată pentru prezentarea la faza inițială a unor concepte, oferind posibilitatea de ghidare a activității elevului în bază de situații practice. Prin intermediul acestei metode se pot reda, prin analogie, diverse situații, raționamente, care pot să reprezinte relații dintre obiecte, fenomene, procese etc. Această metodă se recomandă pentru predarea-învățarea-evaluarea următoarelor unități de conţinut:

- Metode de sortare.
- Pointeri.Variabile pointeri.
- Crearea dinamică a tipului de date tablou. Alocarea dinamică a memoriei.
- Subprograme recursive.

Problematizarea mai poate fi denumită şi predare prin rezolvare de probleme sau predare productivă de probleme. Conform acestei metode instruitul este pus în fata unor dificultăți create în mod deliberat, şi prin depășirea lor învață ceva nou. „Punctul forte" al metodei îl constituie situația-problemă. Din această cauză este necesar de a formula corect situația. La crearea situației de tip problemă se va ține cont de următoarele caracteristici:

A. Situația trebuie să prezinte o dificultate pentru instruit, iar pentru a găsi soluția, acesta se va confrunta cu efort de gândire;
B. Situația trebuie să prezinte interes, astfel încât acesta să acționeze spre a rezolva problema;
C. Situația trebuie să orienteze activitatea instruitului spre a rezolva problema şi de al cointeresa pe acesta de a dobândi noi cunoștințe;
D. Rezolvarea situației nu va fi posibilă fără a apela la resursele recent dobândite.

Prin intermediul situației create, instruitul este cointeresat de a studia, analiza şi a participa la rezolvarea problemei. Aplicarea acestei metode presupune parcurgerea a patru etape:

1. Formularea problemei – este descrisă situația problemă, explicarea, după necesitate a diferitor puncte cheie, care ar permite instruitului să perceapă problema;
2. Studierea problemei – se lucrează în mod independent, sunt reactualizate anumite resurse;
3. Determinarea soluției – în cadrul acestei etape sunt pregătite resursele necesare, se descoperă mijloacele care duc la rezolvarea problemei şi este analizat modul de aplicare a acestora în determinarea soluției;
4. Obținerea rezultatului final – se analizează rezultatul obținut şi sunt formate anumite concluzii.

Această metodă se recomandă pentru predarea-învățarea-evaluarea următoarelor unități de conţinut:

- Tipul de date fișier.
- Structuri.
- Efecte colaterale.
- Supraîncărcarea funcțiilor.

Algoritmizarea reprezintă o metodă de predare-învățare bazată pe utilizarea şi valorificarea algoritmilor în procesul de instruire. Algoritmul de instruire se reprezintă sub forma unui grup de scheme, unui set de operații, iar prin parcurgerea lor într-o ordine bine stabilită duce la rezolvarea unui set de probleme caracteristice unei familii de situații. În rezultatul aplicării acestei metode se va oferi posibilitatea elevului de a elabora treptat propriile scheme, aplicabile în diferite circumstanțe didactice.

Această metodă se recomandă pentru predarea-învățarea-evaluarea următoarelor unități de conţinut:

- Metode de sortare.
- Tablouri de structuri.
- Subprograme recursive.
- Module/Unități de program.

Instruirea asistată de calculator este o metodă didactică care valorifică principiile de modelare şi analiză cibernetică. Prin intermediul calculatorului se pune la dispoziția elevului un set de probleme, care necesită a fi analizate, completate sau elaborate. Utilizarea metodei va oferi posibilitatea de organizare a informației conform cerințelor programei adaptabile la capacitățile fiecărui elev; stimularea cognitivă a elevului prin secvențe didactice şi întrebări ce vizează depistarea unor lacune, probleme, situații-problemă; rezolvarea sarcinilor didactice prezentate anterior prin reactivarea sau obținerea informațiilor necesare de la resursele informatice apelate prin intermediul calculatorului; realizarea unor sinteze recapitulative după parcurgerea unor teme, module de studiu, lecții; asigurarea unor exerciții suplimentare de stimulare a creativității elevului. Această metodă se recomandă pentru predarea-învățarea-evaluarea următoarelor unități de conţinut:

- Metode de sortare;
- Subprograme recursive;
- Programare modulară.

Metoda studiul de caz valorifică o situație reală care se analizează şi se rezolvă. Așa cum problemele rezolvate în stilul orientat pe obiecte au un grad sporit de dificultate, sunt cazuri când este necesar de a prezenta elevului probleme deja rezolvate. Avantajul metodei, constă în faptul că fiecare dintre elev își va aduce aportul la analiza şi rezolvarea problemei. În utilizarea acestei metode se conturează câteva etape: 1) Selectarea şi prezentarea cazului; 2) Organizarea echipelor de lucru; 3) Prelucrarea şi conceptualizarea; 4) Structurarea finală a studiului.

Această metodă se recomandă pentru predarea-învățarea-evaluarea următoarelor unități de conţinut:

- Tipul de date fișier. Operații de specifice tipului de date fișier.
- Metode de sortare.
- Structuri.
- Subprograme.

Instruirea prin proiecte reprezintă o modalitate de instruire grație căreia elevii efectuează o cercetare orientată spre obiective practice şi finalizată într-un produs ce poate fi un obiect, un aparat, o instalație, o culegere tematică, un album, o lucrare științifică etc. Această metodă se recomandă pentru predarea-învățarea-evaluarea următoarelor unități de conţinut:

- Metode de sortare.
- Tablouri de structuri.
- Subprograme.
- Module/ Unități de program.

X. Sugestii de evaluare a competențelor profesionale

Evaluarea competențelor profesionale este procesul prin care sunt colectate şi analizate dovezile necesare pentru judecarea competenței în raport cu cerințele calificării profesionale. Calificarea profesională este documentul în care se descriu rezultatele învățării în concordanță cu cerințele pieței muncii, specificate în standardul ocupațional/ profilul ocupațional. Evaluarea competențelor profesionale este un proces complet diferit de sistemul tradițional de evaluare a cunoștințelor. Evaluarea competențelor profesionale este un proces care presupune consultarea şi colaborarea dintre elev şi profesor. Evaluarea competențelor are loc prin furnizarea de către elev a dovezilor de competență care sunt interpretate de către profesor. Dovezile de competenţă acumulate sunt rezultate considerate parțiale şi atât elevul cât și profesorul pot solicita clarificări suplimentare.

Procedura de evaluare a competențelor profesionale pentru modulul *Programarea procedurală*, va oferi elevilor posibilitatea de a-şi demonstra atât cunoștințele teoretice şi practice. Metodele folosite în procesul de evaluare vor evidenția cunoștințele şi deprinderile necesare pentru efectuarea activităților de muncă şi, mai ales, capacitatea elevului de a obține rezultatele practice așteptate.

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

Evaluarea curentă/formativă se va realiza prin diverse modalități: observarea comportamentului elevului, analiza rezultatelor activității elevului, discuția/conversația, prezentarea proiectelor individuale de activitate. Prin evaluarea curentă/formativă, cadrele didactice informează elevul despre nivelul de performanta; îl motivează să se implice în dobândirea competențelor profesionale.

Evaluarea sumativă se realizează la finele modulului în baza simulării în atelier a unei situații de problemă din contexte profesionale variate, care solicită elevului demonstrarea competenței profesionale. Cadrele didactice

vor elabora sarcini prin care vor orienta comportamentul profesional al elevului spre demonstrarea sistemului de cunoștințe şi abilități. În acest scop, vor fi clar stabiliți indicatorii şi descriptorii de performanță ai procesului şi produsului realizat de către elev.

Portofoliul reprezintă o metodă complexă de evaluare în care un rezultat al evaluării este elaborat pe baza aplicării unui ansamblu variat de probe și instrumente de evaluare. Portofoliul, de regulă este realizat pe o perioadă mai îndelungată (în decursul mai multor ore). Conținutul unui portofoliu este reprezentat de rezultatele la: lucrări practice, studiul individual, investigații, referate și proiecte, observarea sistematică la clasă, autoevaluarea elevului, chestionare de atitudini etc. Alegerea elementelor ce formează portofoliul este realizată de către profesor (astfel încât acestea să ofere informații concludente privind pregătirea, evoluția, atitudinea elevului) sau chiar de către elev (pe considerente de performanță, preferințe etc.). Structurarea evaluării sub formă de portofoliu se dovedește deosebit de utilă, atât pentru profesor, cât și pentru elev sau părinții acestuia. Pentru a realiza o evaluare pe bază de portofoliu, profesorul:

- va comunica elevilor intenția de a realiza un portofoliu, adaptând instrumentele de evaluare ce constituie "centrul de greutate" ale portofoliului la specificul unității de învățare;
- va alege componentele ce formează portofoliul, dând și elevului posibilitatea de a adăuga piese pe care le consideră relevante pentru activitatea sa;
- va evalua separat fiecare piesă a portofoliului în momentul realizării ei, dar va asigura și un sistem de criterii pe baza cărora să realizeze evaluarea globală și finală a portofoliului;
- va pune în evidență evoluția elevului, particularitățile de exprimare și de raportare a acestuia la aria vizată;
- va integra rezultatul evaluării portofoliului în sistemul general de notare.

Competențele elevului se manifestă prin produse concrete, care sunt analizate de către profesor în raport cu aspectele critice stabilite pentru unitate/unitățile de competenţă pentru care este evaluat. Dovezile de competenţă sunt informațiile produse de un elev din care rezultă că îndeplinește toate aspectele descrise de unitatea/unitățile de competenţă pentru care este evaluat, respectiv are cunoștințele şi deprinderile necesare.

Evaluarea nivelului de dezvoltare a competențelor în cadrul orelor:

- ***teoretice*** se va realiza prin teste, exemple de aplicare a cunoștințelor teoretice în practică, machete etc.;
- ***de laborator*** se va realiza prin elaborarea de către elev, în termeni concreți, a aplicațiilor web având la bază unitățile de conținut studiate în cadrul orelor teoretice precum și abilitățile anterior dezvoltate;
- ***de studiu individual*** se va realiza prin studierea de către elev a materialelor suplimentare decât cele oferite în cadrul orelor de tip contact direct și prezentarea de portofolii pentru anumite unități de conținut și aplicații web complexe prin care elevul își va demonstra abilitățile formate.

Probe de evaluare a competențelor, în baza situațiilor de problemă de la viitoarele locuri de muncă:

- elaborarea aplicațiilor de consolă conform specificațiilor propuse;
- evaluarea aplicațiilor de consolă elaborate;
- selectarea structurii de date conform specificațiilor tehnice;
- elaborarea algoritmilor pentru tipurile de date utilizate în procesul de studiu;
- modificarea aplicațiilor de consolă conform specificațiilor propuse;
- testarea aplicațiilor de consolă elaborate.

În calitate de **produse pentru măsurarea competențelor** se vor folosi:

- aplicații de consolă elaborate conform specificațiilor propuse;
- subprograme elaborate conform specificațiilor propuse;
- algoritmi elaborați conform specificațiilor propuse;
- structuri de date gestionate conform specificațiilor propuse.

Criteriile de evaluare a produselor pentru măsurarea competenței vor include:

- Utilizarea corectă a instrucțiunilor limbajului de programare.
- Corectitudinea algoritmilor elaborați.
- Fundamentarea deciziilor.
- Utilizarea terminologiei specifice disciplinei.
- Respectarea termenilor de elaborare a produselor.

Forma finală de evaluare a unității de curs este examenul. Pentru a fi admis la examen elevul trebuie să susțină evaluările curente, să însușească materialul teoretic și să îndeplinească sarcinile propuse pentru studiul individual. Nota finală va fi formată: 60% - media notelor curente și 40% - nota de la examen.

XI. Resursele necesare pentru desfășurarea procesului de studii

Cerințe față de sălile de curs	
Pentru orele teoretice	Sala de clasă dotată cu mobilier școlar și condiții ergonomice adecvate.
Pentru orele de laborator	Laborator de informatică care asigură fiecărui elev o stație de lucru specifică necesităților de instruire.
Cerinţe pentru lecții online	
Pentru orele teoretice și de laborator	Instrumente de comunicare sincronă Platforme de colaborare online Instrumente digitale de evaluare
Cerințe tehnice	
Parametri tehnici minimi ale calculatorului	**Procesor**: 2 GHz **Memorie operativă**: 4 GB **Unitate de stocare**: 500 GB **Afișaj și grafică**: size: 22'', resolution: 1366x768 **Network**: Ethernet, 100 Mb
Software	Sistem de Operare Microsoft Windows Code::Blocks Dev C/CPP Visual Studio Compilator online https://www.onlinegdb.com/online_c++_compiler

XII. Resursele didactice recomandate elevilor

Nr. crt.	Denumirea resursei	Locul în care poate fi consultată/ accesată resursa
1.	A. Ruceanu, Suport de curs: *Proiectarea algoritmilor.*	http://www.runceanu.ro/adrian/wp-content/cursuri/pa2014.php
2.	A. Deaconu, Ghid pentru studenți din domeniul *Informatică*, 2010.	dokumen.tips/documents/adrian-deaconu-programare-procedurala.html
3.	E. Cerchez, M. Șerban, *Programarea în limbajul C/C++* pentru liceu. Editura Polirom, Iași, 2005.	https://www.academia.edu/29418923/Emanuela_Cerchez_Marinel_Serban_Programare_Liceu_c_c_1
4.	G. Vasilache, S. Gîncu. *Culegere de probleme la informatica*, Chișinău, 2012.	http://en.calameo.com/read/002801569a611d413be1c
5.	Suport de curs: *Elemente de algoritmică şi limbaje de programare*.	https://ro.scribd.com/document/48218378/21297279-programare-procedurala

Tutoriale pe Internet

1. http://www.programming.com
2. http://www.cplusplus.com
3. http://www.infoarena.ro/