



Ministerul Educației al Republicii Moldova
Centrul de Excelență în Energetică și Electronică

"Aprob"
Directorul Centrului de Excelență în
Energetică și Electronică,
Vrînceanu
2017

Curriculumul modular
S.07.O.021 Limbaje de programare

Specialitatea: 71420 Automatizarea proceselor tehnologice
Calificarea: Tehnician automatizare a proceselor de producție

Chișinău 2017

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:

1. *Romanciuc Vera, grad didactic doi, Centrul de Excelență în Energetică și Electronică.*
2. *Tofan Tatiana, grad didactic unu, Centrul de Excelență în Energetică și Electronică*

Aprobat de:

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

Director

Vasile VRÎNCEANU

V. Vrînceanu
17 Aprilie 2017



Recenzenți:

1. Alexandru Corețchi, Administrator NET&PULS SRL.
2. Denis Țapotei, șef secție "Automatică și aparate de măsură și control" Fabrica S.A. "Bucuria"
- 3.

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	5
IV. Administrarea modulului.....	5
V. Unitățile de învățare.....	6
VI. Repartizarea orientativă a orelor pe unități de învățare.....	9
VII. Studiu individual ghidat de profesor	9
VIII. Lucrările de laborator recomandate	11
IX. Sugestii metodologice	12
X. Sugestii de evaluare a competențelor profesionale	12
XI. Resursele necesare pentru desfășurarea procesului de studii	15
XII. Resursele didactice recomandate elevilor.....	15

I. Preliminarii

Modulul "Limbaje de programare" prezintă noțiuni fundamentale de programare folosind limbajele Assembler și C pentru a programa pe viitor microcontrolere din diferite familii. Limbajele Assembler și C sunt ușor de utilizat cu interfețe de lucru explicite.

Modulul respectiv este structurat în patru unități de învățare. Primele două unități de învățare se referă la noțiuni generale ale limbajului Assembler, utilizarea instrucțiunilor și funcțiilor limbajului Assembler. Unitățile de învățare 3 și 4 se referă la noțiuni generale ale limbajului C, utilizarea instrucțiunilor și funcțiilor limbajului C

Statutul Curriculumului. Curriculumul modular "Limbaje de programare" este un document normativ și obligatoriu pentru realizarea procesului de pregătire a tehnicienilor în învățământul profesional tehnic postsecundar, 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 și certificare în contextul unei pedagogii axate pe competențe;
- reper pentru proiectarea didactică și desfășurarea procesului educațional din perspectiva unei pedagogii axate pe competențe;
- componentă de bază pentru elaborarea strategiei de evaluare și certificare;
- orientare a procesului educațional spre formare de competențe la elevi;
- componentă fundamentală pentru elaborarea manualelor tipărite, manualelor electronice, ghiduri metodologice, testelor de evaluare.

Beneficiarii Curriculumului. Curriculumul este destinat:

- profesorilor din instituțiile de învățământ profesional tehnic postsecundar;
- autorilor de manuale și ghiduri metodologice;
- elevilor care își fac studiile la specialitatea Automatizarea proceselor tehnologice;
- membrilor comisiilor de identificare, evaluare și recunoaștere a rezultatelor învățării, dobândite în contexte non-formale și informale.

Scopul modulului este de a transmite elevilor noțiunile fundamentale privind programarea în Assembler și C și de a le forma deprinderi practice de lucru în cadrul unui mediu de programare.

Pentru studierea cu succes a acestui modul este necesar de parcurs următoarele unități de curs:

- informatica curriculară;
- F.06.O.014 Analiza și sinteza circuitelor numerice

II. Motivația, utilitatea modulului pentru dezvoltarea profesională

Modulul "Limbaje de programare" contribuie la formarea competențelor profesionale ale tehnicianului în domeniul automatizării necesare pentru îndeplinirea atribuțiilor și sarcinilor de lucru stabilite în calificarea profesională:

- cunoștințe factice, principii, procese și concepte generale din domeniul elaborării produselor program;

- abilități cognitive și practice necesare pentru elaborarea programelor conform tematicilor incluse;
- asumarea responsabilității pentru mentenanța de aplicații.
- Competențele de dezvoltare în cadrul acestui modul vor fi necesare pentru studierea unităților de curs orientate spre elaborarea/dezvoltarea produselor program. De asemenea, ele vor fi de un real folos în activitatea profesională a tehnicianului în programarea microcontrolerelor din diferite familii.

III. Competențele profesionale specifice modului

CS1. Elaborarea programelor în Assembler.

CS2. Asigurarea funcționalității programelor în Assembler, conform specificațiilor tehnice.

CS3. Elaborarea programelor în C.

CS4. Asigurarea funcționalității programelor în C, conform specificațiilor tehnice.

IV. Administrarea modului

Semestrul	Numărul de ore			Modalitatea de evaluare	Numărul de credite	
	Total	Contact direct				
		Prelegeri	Practică/ Seminar	Lucrul individual		
V	120	20	20	80	examen	4

V. Unitățile de învățare

Unități de competență	Unități de conținut	Abilități
1. Arhitectura 80x86. Moduri de adresare		
UC1. Utilizarea modurilor de adresare în limbajul ASSEMBLER	<ol style="list-style-type: none"> 1. Lexicul, directivele de determinare a datelor, <ul style="list-style-type: none"> - Identificatori - Numere întregi - Caractere de date - Comentarii - Comenzi - Etichete - Operanzi - Directiva byte - Directiva word - Directiva double word - Directiva quade word 2. Registrele. Moduri de adresare. <ul style="list-style-type: none"> - Tipuri de registre - Stiva - Registrul fanionilor - Adresarea directă - Adresarea indirectă 	<ol style="list-style-type: none"> A1. Scrierea identificatorilor. A2. Declararea variabilelor/constantelor conform tipului de dată solicitat. A3. Scrierea unui comentariu. A4. Inițializarea variabilelor/constantelor cu valori prestabilite. A5. Determinarea directiveilor. A6. Selectarea operanzilor în dependență de specificul tipului de dată utilizat. A7. Identificarea operanzilor. A8. Determinarea tipurilor de registre. A9. Identificarea fanionilor. A10. Realizarea adresărilor directe și indirecte. A11. Identificarea părților componente ale unui program.

Unități de competență	Unități de conținut	Abilități
2. Setul de instrucțiuni 8086		
UC2. Utilizarea programelor de navigare în Internet.	2. Browser-e Web. 3. Setările browser-ului. 4. Extensiile browser-ului. 5. Certificate de securitate. 6. Instrumentele dezvoltatorului Web.	A12. Lansarea/închiderea browser-ului Web. A13. Identificarea elementelor de interfață ale browser-ului Web. A14. Modificarea setărilor browser-ului Web. A15. Instalarea și deinstalarea extensiilor browser-ului Web. A16. Verificarea certificatelor de securitate ale browser-ului Web. A17. Instalarea certificatelor de securitate ale browser-ului Web. A18. Identificarea instrumentelor dezvoltatorului Web integrate în browser-ul Web.
3. Tipuri de date simple. Instrucțiuni		
UC3. Elaborarea programelor simple în limbajul C;	1. Inițiere în limbajul C. Vocabularul și sintaxa limbajului. Tipuri de date: - vocabularul limbajului C; - funcția printf(); - funcția scanf(); - operatori în limbajul C. 2. Instrucțiuni de ramificare: - Instrucțiunea if; 3. Instrucțiuni repetitive: - Instrucțiunea for;	A19. Scrierea simbolurilor specifice în limbajul C. A20. Identificarea cuvintelor cheie în limbajul C. A21. Scrierea identificatorilor. A22. Declararea variabilelor/constantelor conform tipului de dată solicitat. A23. Inițializarea variabilelor/constantelor cu valori prestabilite A24. Utilizarea funcțiilor de scriere și citire în limbajul C. A25. Selectarea operatorilor în dependență de specificul tipului de dată utilizat. A26. Identificarea părților componente ale unei instrucțiuni de ramificare. A27. Utilizarea instrucțiunilor de ramificare. A28. Identificarea părților componente ale unei instrucțiuni repetitive. A29. Implimentarea algoritmilor repetitivi în limbajul de programare C.

Unități de competență	Unități de conținut	Abilități
4. Tipuri structurate de date		
UC4. Elaborarea programelor cu ajutorul tipurilor structurate de date	1. Masive în limbajul C. Masive unidimensionale și bidimensionale: <ul style="list-style-type: none"> - Tipuri de date tablou - Noțiunea de tablou. - Declararea unui tablou într-un program. - Tablouri unidimensionale și bidimensionale; 2. Șiruri de tip caracter <ul style="list-style-type: none"> - Declararea tipului șir de caractere - Operații cu șiruri de caractere - Funcții standard pentru operații cu șiruri de caractere 4. Structuri de program	A30. Declararea tablourilor unidimensionale și bidimensionale. A31. Inițializarea și citirea elementelor unui tablou unidimensional/bidimensional. A32. Afișarea, permutarea și eliminarea elementelor dintr-un tablou. A33. Declararea și utilizarea șirurilor de caractere. A34. Copierea și concatenarea șirurilor de caractere. A35. Căutarea unui șir de caractere. A36. Utilizarea funcțiilor standard la prelucrarea șirurilor de caractere A37. Utilizarea funcțiilor standard la programarea și utilizarea structurilor

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.	Arhitectura 80x86. Moduri de adresare	16	4		12
2.	Setul de instrucțiuni 8086	36	6	8	22
3.	Tipuri de date simple. Instrucțiuni	24	6	4	14
4.	Tipuri de date structurate	44	4	8	32
	Total	120	20	20	80

VII. Studiu individual ghidat de profesor

Materii pentru studiul individual	Produse de elaborat	Modalități de evaluare	Termeni de realizare
1. Arhitectura 80x86. Moduri de adresare			
Vocabularul și sintaxa de bază a limbajului ASSEMBLER	Referat	Comunicare	2 ore
Determinarea fanionilor	Lucrare individuală Calculul binar și determinarea fanionilor	Prezentarea lucrării	2 ore
Crearea unui început de program în ASSEMBLER	Lucrare individuală Cu ajutorul lexicului de elaborat antetul unui program	Prezentarea lucrării	2 ore
Interfața mediului de programare	Studiu de caz	Demonstrare la calculator	2 ore
Descrierea pachetului de programe Turbo Assembler	Studiu de caz Comenzi folosite în Turbo Assembler	Demonstrare la calculator	2 ore
2. Setul de instrucțiuni 8086			
Instrucțiuni de transfer ce operează cu stiva.	Referat	Comunicare	2 ore
Alte tipuri de transfer	Referat	Comunicare	2 ore
Instrucțiuni aritmetice de înmulțire	Lucrare individuală	Demonstrare la calculator	2 ore
Instrucțiuni aritmetice de	Lucrare individuală	Demonstrare	2 ore

Materii pentru studiul individual	Produse de elaborat	Modalități de evaluare	Termeni de realizare
împărțire		la calculator	
Instrucțiuni de negare	Lucrare individuală	Demonstrare la calculator	2 ore
Instrucțiuni logice SAU EXCLUSIV	Lucrare individuală	Demonstrare la calculator	2 ore
Instrucțiuni de deplasare logică la dreapta	Lucrare individuală	Demonstrare la calculator	2 ore
Instrucțiuni de deplasare aritmetica la stînga	Lucrare individuală	Demonstrare la calculator	2 ore
rotație simplă stînga	Lucrare individuală	Demonstrare la calculator	2 ore
rotație ciclică dreapta	Lucrare individuală	Demonstrare la calculator	2 ore
Salt necondiționat	Lucrare individuală	Demonstrare la calculator	2 ore
3. Tipuri de date simple. Instrucțiuni			
Alfabetul limbajului C	Referat	Comunicare	2 ore
Tipuri fundamentale de date în C	Referat	Comunicare	2 ore
Instrucțiuni de ramificare IF-ELSE	Lucrare individuală	Demonstrare la calculator	2 ore
Instrucțiunea de ramificare switch	Lucrare individuală	Demonstrare la calculator	2 ore
Instrucțiunea Break	Lucrare individuală	Demonstrare la calculator	2 ore
Instrucțiunea while	Lucrare individuală	Demonstrare la calculator	2 ore
Instrucțiunea do-while	Lucrare individuală	Demonstrare la calculator	2 ore
4. Tipuri de date structurate.			
Initializarea unui tablou unidimensional cu valori cunoscute	Lucrare individuală	Demonstrare la calculator	2 ore
Inițializarea unui tablou unidimensional cu valori necunoscute.	Lucrare individuală	Demonstrare la calculator	2 ore
Găsirea unui element dintr-un tablou unidimensional	Lucrare individuală	Demonstrare la calculator	2 ore

Materii pentru studiul individual	Produse de elaborat	Modalități de evaluare	Termeni de realizare
Inițializarea unui tablou bidimensional cu valori cunoscute	Lucrare individuală	Demonstrare la calculator	2 ore
Inițializarea unui tablou bidimensional cu valori necunoscute.	Lucrare individuală	Demonstrare la calculator	2 ore
Găsirea unui element dintr-un tablou bidimensional	Lucrare individuală	Demonstrare la calculator	2 ore
Declararea masivului de tip char	Lucrare individuală	Demonstrare la calculator	2 ore
Introducerea unui șir și afișarea șirului	Lucrare individuală	Demonstrare la calculator	2 ore
Căutarea primei apariții a unui caracter în șir	Lucrare individuală	Demonstrare la calculator	2 ore
Declararea și descrierea structurii	Prezentare	Comunicare	2 ore
Inițierea variabilelor de tip structură	Lucrare individuală	Demonstrare la calculator	2 ore
Portofoliul	Prezentarea portofoliului	Comunicare	4 ore
Inițializarea unui tablou unidimensional cu valori cunoscute	Lucrare individuală	Demonstrare la calculator	2 ore
Inițializarea unui tablou unidimensional cu valori necunoscute.	Lucrare individuală	Demonstrare la calculator	2 ore

VIII. Lucrările de laborator recomandate

1. Reprezentarea datelor în memoria calculatorului
2. Programe liniare
3. Programe cu ramificări și bucle
4. Subprograme și transferul parametrilor
5. Elaborarea și programarea algoritmilor de structură ramificată (IF)
6. Elaborarea și programarea algoritmilor de structură ciclică iterativă (FOR)
7. Programarea și utilizarea tablourilor unidimensionale
8. Programarea și utilizarea tablourilor bidimensionale
9. Programarea și utilizarea datelor de tip șir de caractere
10. Programarea și utilizarea structurilor

IX. Sugestii metodologice

Curriculumul la modulul "Limbaje de programare" are drept scop formarea și dezvoltarea competențelor profesionale ale elevilor din învățământul profesional tehnic postsecundar. Dinamica domeniului tehnologiilor informaționale este extrem de rapidă, impune actualizarea permanentă a produselor hardware și software prin prezentarea celor mai noi versiuni ale mediilor de dezvoltare a aplicațiilor, tehnologiilor și platformelor de programare, astfel încât absolvenții mai ușor să se adapteze evoluțiilor ulterioare pe piața muncii.

Locul de desfășurare a activităților de învățare se recomandă a fi o sală echipată cu o 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. Se propune utilizarea metodelor de predare-învățare activ-participative, printre avantajele cărora putem enumera următoarele:

- sunt centrate pe elev și activitate;
- pun accent pe dezvoltarea gândirii, formarea aptitudinilor și a deprinderilor;
- încurajează participarea elevilor, inițiativa, implicarea și creativitatea;
- determină un parteneriat profesor-elev prin realizarea unei comunicări multidirecționale;

Se recomandă orientarea către metode bazate pe rezolvarea unor sarcini de lucru, utilizându-se cu precădere rezolvarea unei game cât mai variate de aplicații practice și punându-se accent pe realizarea cu exactitate și la timp a cerințelor sarcinilor de lucru. Realizarea proiectelor în cadrul activităților practice va urmări dezvoltarea abilităților de lucru în echipă.

Se vor alege cele mai potrivite metode didactice: descoperire, discuția în grup, dezbateră/masa rotundă, studiul de caz, observația individuală. Specificul modulului impune metode didactice interactive, recomandând mai ales învățarea prin metode practice/activități de laborator, proiecte, portofoliul electronic. În activitățile individuale, accentul se va pune pe studiere, analiza și sistematizarea materialului teoretic și practic în scopul îndeplinirii sarcinilor de lucru individual. Acestea vor fi prezentate în formă de portofolii, proiecte, sarcini specifice etc.

X. Sugestii de evaluare a competențelor profesionale

Evaluarea competențelor profesionale este procesul prin care sunt colectate și 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 și colaborarea dintre elev și profesor. Evaluarea competențelor are loc prin furnizarea de către elev a dovezilor de competență care sunt interpretate de către profesor.

Dovezile de competență acumulate sunt rezultate considerate parțiale și atât elevul cât și profesorul pot solicita clarificări suplimentare.

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

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. 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.

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 performanță; 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 și abilități. În acest scop, vor fi clar stabiliți indicatorii și descriptorii de performanță ai procesului și 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 forma 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 competență pentru care este evaluat. Dovezile de competență sunt informațiile produse de un elev din care rezultă că îndeplinește toate aspectele descrise de unitatea/unitățile de competență pentru care este evaluat, respectiv are cunoștințele și 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.
- Ținuta lingvistică.
- Respectarea termenilor de elaborare.

XI. Resursele necesare pentru desfășurarea procesului de studii

Cerințe față de sălile de curs	
Pentru orele teoretice	Cabinet de informatică cu 12-14 calculatoare. Tablă interactivă.
Pentru orele de laborator	Laborator de informatică care asigură fiecărui elev un calculator. Tablă interactivă.
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. Assembler emu86.exe. Turbo C. Notepad, Office. Browser: Chrome.

XII. Resursele didactice recomandate elevilor

Nr. crt.	Denumirea resursei	Locul în care poate fi consultată/ accesată/ procurată resursa	Numărul de exemplare disponibile
1.	Tutorial Limbaj de Asamblare (Assembler) Intel 8086 – Partea 1 – Elemente de baza	http://www.itcsolutions.eu/2010/02/04/tutorial-limbaj-de-asamblare-assembler-intel-8086-%E2%80%93-parte-1-%E2%80%93-elemente-de-baza/	
2.	Tutorial Limbaj de Asamblare (Assembler) Intel 8086 – Partea 2 – Prima aplicatie	http://www.itcsolutions.eu/2010/02/04/tutorial-limbaj-de-asamblare-assembler-intel-8086-%E2%80%93-parte-2-%E2%80%93-prima-aplicatie/	
3.	Programarea în Limbaj de Asamblare Îndrumător de Laborator	http://users.utcluj.ro/~ancapop/labpla/asm_labs.pdf	
4.	Inițiere în limbajul C, Ștefan Buzurniuc, Chișinău, Evrica 2004	CEEE	20
5.	Introducere în C B. Kernighan, D. Richie-Limbajul C	http://www.seap.usv.ro/~valeriu/lupu/Cap2.pdf	
6.	PROGRAMAREA CALCULATORILOR în limbajul C, Florian Moraru	http://andrei.clubcisco.ro/cursuri/f-f-sym/1pc/2009CC/cursuri/Moraru-PC2008.pdf	