

FIȘA DISCIPLINEI

Universitatea Academiei de Științe a Moldovei Facultatea Științe exacte			Denumirea cursului: Arhitectura calculatorului și sisteme de operare Codul cursului în planul de studii: F.03.O.016				
Nivelul calificării ISCED: 6 Domeniul de formare profesională: 443 Matematică Specialitatea: 443.1 Matematică			Catedra responsabilă de curs: Matematică și Informatică Titular/Responsabil de curs: Gladei Anatolie, lector superior				
Total ore			Număr de ore pe tipuri de activități			Forma de evaluare	Număr de credite
total	contact direct	studiu individual	curs	seminar	laborator		
120	60	60	30	30	0	E	4

Descrierea succintă a corelării cursului cu programul de studii

Arhitectura calculatorului și sisteme de operare este o disciplină de specialitate, care asigură pregătirea studenților în domeniul informaticii. Cursul este adresat studenților anului II, sem. I specialitățile “Informatica”, “Informatică aplicată” și “Matematică”. Arhitectura calculatoarelor este cercetată din diferite puncte de vedere distincte, fiecare fiind interpretat ca un nivel al ierarhiei sistemului de calcul.

La lecțiile practice se vor analiza exemple tipice de algoritmi elementari, realizarea lor în limbajul de asamblare. Vor fi folosite translatoarele TASM, MASM editoarele de legături interne TLINK și depanatorul TD. Se va folosi linia de comanda pentru dialogul cu sistemul de calcul. Vor fi cercetate structurile sistemelor de operare și a sistemelor de fișiere.

Competențe dezvoltate în cadrul cursului

Competențe generale:

- Cunoștințe de bază în domeniul matematicii elementare și a matematicii superioare;
- capacitate de analiză și sinteză a teoriilor, metodelor și datelor provenite din diverse compartimente ale matematicii sau din surse adiacente;
- capacitate de a aplica cunoștințele teoretice la studiul problemelor practice;
- abilitate de a studia independent diverse surse și capacitate de a continua studiile cu un grad sporit de autoinstruire;
- capacitate de aplicare a tehnologiilor informaționale și a cel puțin unei limbi străine în studiu și cercetare;
- capacitatea de a lucra atât independent, cât și în echipă, în funcție de cerințele de moment.

Competențe specifice:

- dezvoltarea capacității de memorare, generalizare și analiză critică a informației care permite viitorului specialist să-și asume responsabilități și să se adapteze exigențelor societății;
- cunoașterea și aplicarea metodologiei contemporane de cercetare, a ideilor, algoritmilor sau modelelor matematice în soluționarea problemelor practice;
- argumentarea importanței investigațiilor reieșind din direcțiile de cercetare în domeniul matematicii teoretice și aplicative;
- analiza metodelor de rezolvare a problemelor și a soluțiilor în concordanță cu datele din literatura de specialitate;
- comunicarea logică, coerentă, argumentată a informației, ideilor, soluțiilor în limba de stat și alte limbi, în medii profesionale și alte medii social-economice;
- identificarea domeniilor și argumentarea avantajelor pe care le oferă matematica în rezolvarea unor probleme profesionale, relevante pentru societate;
- formarea abilităților de a însuși metode noi și mecanisme eficiente de soluționare a problemelor și de implementare a rezultatelor în activitatea profesională;
- dezvoltarea capacității de a studia individual diverse surse informaționale care contribuie la sporirea gradului de autoinstruire în domeniul științei.

Finalități de studii ale cursului

la nivel de aplicare studenții vor:

- aplica particularitățile translatorului din limbajul Assembler și ale editorului de legături (TASM, MASM, TLINK, LINK);
- utiliza instrucțiunile limbajului Assembler;
- utiliza procedeele de programare în limbajul de asamblare, întreruperii și mecanismului de prelucrare al lor.

la nivel de integrare studenții vor:

- utiliza gândirea algoritmică în soluționarea claselor de probleme din matematica economie, informatică ș.a.;
- aprecia gradul de complexitate al problemelor și alegerea sau elaborarea de algoritmi efectivi pentru soluționarea lor;
- elaborează algoritmul de soluționare a problemelor;
- determină resursele hard și soft pentru realizarea algoritmului de soluționare al problemei;
- percepe formalizarea limbajului natural;
- propune sfere de aplicare ale algoritmilor și diferitor tipuri de structuri de date;
- transferă modelele de raționament în cadrul altor domenii;
- stabilește erori de raționament.

Condiții prerechizite: pentru studierea cursului *Arhitectura calculatorului și sisteme de operare* este necesară inițierea în Tehnologii informaționale și bazele programării, Algoritmi și programare, Programarea distribuită – platforma Java.

Teme de bază: Calculatoare electronice. Microprocesoare. Programarea microprocesoarelor. Procesoare pe 16, 32 și 64 biți. Regimul real și regimul protejat de lucru al microprocesorului. Arhitectura de bază. Setul de registre al microprocesorului. Organizarea memoriei. Formarea adresei fizice în regim real. Formatul instrucțiunilor. Codificarea instrucțiunilor. Moduri de adresare a operanzilor. Noțiunile fundamentale ale limbajului de asamblare. Definierea și utilizarea segmentelor. Sistemul de întreruperi. Reprezentări interne ale datelor. Definierea și inițializarea datelor în limbaj de asamblare. Instrucțiuni de transfer a datelor Instrucțiuni aritmetice. Instrucțiuni logice. Instrucțiuni de transfer al controlului. Legătura între limbajul de asamblare și limbajele Pascal și C. Sisteme de operare. Clasificarea sistemelor de operare. Sisteme de fișiere. Nucleul sistemului de operare. Funcțiile sistemelor de operare. Sistemul de comenzi al Sistemului de operare. Interfețe grafice.

Strategii de predare-învățare: prelegeri, lucrări individuale, consultații.

Strategii de evaluare: teste de evaluare, prezentări, rapoarte, dezbateri, elaborarea portofoliilor, teze/proiecte etc. Nota finală se constituie din rezultatul evaluării finale (40%), curente (40 %) și calității lucrului individual al studentului pe parcursul semestrului (20%).

Bibliografie selectivă:

1. Peter Norton. Secrete PC // Traducere din engleză . București. Teora. 1996;
2. Vasile Lungu. Procesoare Intel Programare în Limbaj de asamblare. București Teora. 2000;
3. Gheorge Muscă. Programare în limbaj de asamblare. București. Teora. 1998;
4. В. Юров, С. Хорошенко. Assembler Учебный курс. Санкт-Петербург. Питер. 2003;
5. Л . Скэнвон. Персональные ЭВМ IBM PC и XT. Программирование на языке ассемблера. \ \ Перевод с англ. - Москва . Радио и Связь. 1989;
6. С.Зубков. Assembler для DOS, Windows и UNIX. М., ДМК. 2000;
7. Кип Р. Ирвин. Язык Ассемблера для процессоров Intel. М., Вильямс. 2002;
8. Питер Абель. Язык Ассемблера для IBM PC и программирования. М., высшая школа, 2004;
9. Г.Майко. Ассемблер для IBM PC. М., Бизнес-Информ, 1997.

Data

Semnătura