



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Nazwa przedmiotu		Kod ECTS	
Architektura komputerów		11.3.0787	
Nazwa jednostki prowadzącej przedmiot			
null			
Studia			
wydział	kierunek	poziom	pierwszego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
dr hab. Piotr Szuca			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		6 Przedmiot w wymiarze 30h wykładu i 30h ćw. lab. + praca własna studenta	
Wykład, Ćw. laboratoryjne			
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Ćw. laboratoryjne: 30 godz., Wykład: 30 godz.			
Cykl dydaktyczny			
2016/2017 zimowy			
Status przedmiotu		Język wykładowy	
obowiązkowy		polski	
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne	
<ul style="list-style-type: none"> - Metoda projektów (projekt badawczy, wdrożeniowy, praktyczny) - Rozwiązywanie zadań - Wykład konwersatoryjny - Wykład problemowy - Wykład z prezentacją multimedialną 		Sposób zaliczenia	
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - egzamin pisemny z pytaniami (zadaniami) otwartymi - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru 	
		Podstawowe kryteria oceny	
		<p>Przedmiot kończy się egzaminem pisemnym składającym się z części teoretycznej (test) i praktycznej (program do napisania w języku assembler). Aby zaliczyć przedmiot, należy z każdej z części egzaminu otrzymać przynajmniej 50% punktów. Liczba punktów, które można otrzymać z części teoretycznej egzaminu jest równa liczbie punktów, które można otrzymać z części praktycznej egzaminu.</p> <p>Aby zostać dopuszczonym do egzaminu konieczne jest zaliczenie ćwiczeń laboratoryjnych, O zaliczeniu ćwiczeń decyduje sumaryczna liczba punktów otrzymanych za wykonywane w trakcie zajęć projekty.</p>	
Sposób weryfikacji założonych efektów kształcenia			

zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność w dyskusji	obserwacja
Wiedza							
K_W02	x		x				
K_W05	x		x				
K_W12							x
Umiejętności							
K_U01	x		x				
K_U03			x				
K_U04			x				
K_U05			x				
K_U06	x		x				
K_U07	x		x				

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi**A. Wymagania formalne****B. Wymagania wstępne**

Umiejętność programowania w języku C.

Cele kształcenia

1. Przedstawienie budowy, działania i programowania w języku assembler procesora IAPx86.
2. Przedstawienie ogólnej budowy i charakterystyki poszczególnych bloków funkcjonalnych komputera PC/XT/AT/ATX.
3. Wprowadzenie do elementów techniki cyfrowej.

Treści programowe

Kodowanie informacji. Alfabety komputerowe: ASCII, strony kodowe CPxxx, CPxxxx, ISO8859*, Unicode.

Pozycyjny zapis liczb. System dziesiętny, szesnastkowy, oktalny, binarny.

Elementy techniki cyfrowej. Układy kombinacyjne. Bramki AND,OR,NOT,NAND,NOR,XOR. Postać kanoniczna funkcji. Minimalizacja funkcji logicznej metodą tablic Karnaugh.

Elementy techniki cyfrowej. Układy sekwencyjne. Przerzutniki RS, Latch, D, JK-MS.

Procesor. Model logiczny, rejestr, akumulator, licznik rozkazów, ALU, lista rozkazów.

Procesor IAPx86 w trybie rzeczywistym. Segmentacja pamięci w trybie rzeczywistym.

Podstawy języka assembler. Struktura programu, podstawowe dyrektywy, makra.

Stos procesora i jego użycie w programach.

Budowa procesora IAPx86 w trybie chronionym. Stronicowanie. Poziomy ochrony. Wielozadaniowość.

Przerwania programowe, sprzętowe i wyjątki.

Budowa komputera PC/XT/AT/ATX. Model komputera, CPU, pamięć, urządzenia we/wy.

Blok funkcjonalne komputera PC: CTC, DMA, PIC, RTC, kontroler klawiatury, ROM systemowy i jego zasoby.

Interfejsy komputera PC: RS232C, IEEE1284, USB, ATA, SATA.

Wykaz literatury

- P. Metzger, A. Jelowiecki, "Anatomia PC", Helion.
 A. Skorupski, "Podstawy budowy i działania komputerów", WKŁ.
 B.S. Chalk, "Organizacja i architektura komputerów", WNT.
 S. Kruk, "Procesor Pentium.", PLJ.
 J. Biernak, "Metody i układy arytmetyki komputerowej.", PWN.
 D.W. Lewis, "Między assemblerem a językiem C.", Wydawnictwo RM.

Efekty kształcenia (obszarowe i kierunkowe)

K_W02: ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie programowania, algorytmów i złożoności, architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, języków i paradygmatów programowania, baz danych, inżynierii oprogramowania, języków formalnych;
 K_W05: ma podstawową wiedzę na temat architektury systemów cyfrowych;
 K_W12: zna podstawowe zasady bezpieczeństwa i higieny

Wiedza

Student wie jak wygląda budowa (z punktu widzenia programisty) procesora IAPx86 zarówno w trybie rzeczywistym jak i chronionym. Wie na czym polegają mechanizmy segmentacji pamięci, stronicowania i ochrony zasobów. Wie na czym polega mechanizm przerwań dostępny w procesorze i zaimplementowany w komputerze PC/XT/AT/ATX. Wie w jaki sposób, na poziomie warstwy sprzętowej kodowana jest informacja. Zna podstawowe bloki funkcjonalne komputera PC. Zna podstawowe grupy instrukcji procesora (adresowanie, instrukcje arytmetyczno-logiczne).

<p>pracy w zawodzie informatyka;</p> <p>K_U01: potrafi zastosować wiedzę matematyczną do formułowania, analizowania i rozwiązywania prostych zadań związanych z informatyką;</p> <p>K_U03: potrafi pracować indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów;</p> <p>K_U04: potrafi porozumiewać się przy użyciu różnych technik w środowisku zawodowym oraz w innych środowiskach, w tym w języku angielskim oraz z wykorzystaniem narzędzi informatycznych;</p> <p>K_U05: potrafi pisać, uruchamiać i testować programy w wybranym środowisku programistycznym;</p> <p>K_U06: projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy; wykorzystuje podstawowe techniki algorytmiczne i struktur danych;</p> <p>K_U07: rozumie niskopoziomowe zasady wykonywania programów.</p>	<p>Umiejętności</p> <p>Student umie napisać w języku assembler proste programy manipulujące różnymi sposobami prezentacji informacji (w tym przedstawienie informacji w systemie binarnym, szesnastkowym, oktalnym, dziesiętnym). Potrafi pisać programy wykorzystujące koprocesor arytmetyczny i programować z użyciem procedur. Potrafi wyjaśnić na czym polegają różne tryby adresowania, segmentacja w trybie rzeczywistym i chronionym.</p> <p>Potrafi omówić działanie i przeznaczenie poszczególnych bloków funkcjonalnych komputera PC/XT/AT/ATX.</p> <p>Potrafi wyjaśnić różnicę między przerwaniem sprzętowymi a programowymi.</p> <p>Kompetencje społeczne (postawy)</p>
<p>Kontakt</p> <p>piotr.szuca@mat.ug.edu.pl</p>	