


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 systemów komputerowych (Z)		11.3.2110	
Nazwa jednostki prowadzącej przedmiot			
Instytut Matematyki			
Studia			
wydział	kierunek	poziom	pierwszego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	niestacjonarne (zaoczne)
		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		3 Przedmiot w wymiarze 10h wykładu i 10h ćw. lab. + praca własna studenta 55h	
Wykład, Ćw. laboratoryjne			
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Wykład: 10 godz., Ćw. laboratoryjne: 10 godz.			
Termin realizacji przedmiotu			
2024/2025 letni			
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"> - Rozwiązywanie zadań - Sporządzanie i uruchamianie programów komputerowych - Wykład problemowy - Wykład z prezentacją multimedialną 		Sposób zaliczenia	
		Zaliczenie na ocenę	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - wykonanie prac zaliczeniowych w trakcie trwania semestru - kolokwium 	
		Podstawowe kryteria oceny	
		Sposób oceniania	Próg zaliczeniowy
		Prace zaliczeniowe wykonywane w trakcie semestru	50%
		Kolokwium	50%
		Obserwacja postawy studenta	100%
		Składowa oceny końcowej	
			50%
			50%
			0%
Sposób weryfikacji założonych efektów uczenia się			

zakładany efekt kształcenia	prace zaliczeniowy wykonywane w trakcie semestru	kolokwium	obserwacja postawy studenta
Wiedza			
K_W07		+	
P_W1		+	
P_W2		+	
P_W3		+	
Umiejętności			
K_U08	+		
P_U1	+		
P_U2	+		
Kompetencje			
K_K01			+
K_K03			+
P_K1			+
P_K2			+

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

Brak

B. Wymagania wstępne

Umiejętność programowania w języku C.

Cele kształcenia

1. Przedstawienie budowy, działania i programowania w języku assembler procesora IAPx86.
2. Omówienie wybranych modułów jądra systemu operacyjnego (obsługa procesów, pamięci).

Treści programowe

Kodowanie informacji liczbowych i tekstowych.
 Pozycyjny zapis liczb. System dziesiętny, szesnastkowy, binarny.
 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.

Wykaz literatury

P. Metzger, A. Jełowicki, "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.

Kierunkowe efekty uczenia się

K_W07: zna najważniejsze elementy architektury systemów komputerowych oraz zasady działania systemów operacyjnych ze szczególnym uwzględnieniem współbieżności, szeregowania zadań i zarządzania pamięcią oraz procesami
 K_U08 : korzysta z zaawansowanych funkcjonalności systemów operacyjnych, w szczególności związanych z aspektami sieciowymi
 K_K01 : zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się

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. Wie w jaki sposób, na poziomie warstwy sprzętowej kodowana jest informacja. Zna podstawowe grupy instrukcji procesora (adresowanie, instrukcje arytmetyczno-logiczne).
 Efekty przedmiotowe:
 P_W1: zna podstawowe polecenia języka assembler (adresowanie, instrukcje arytmetyczno-logiczne) (K_W07)

<p>K_K03 : potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych</p>	<p>P_W2: zna konwencje przekazywania argumentów do procedur, sposoby przechowywania zmiennych globalnych i lokalnych w programach (ABI) (K_W07) P_W3: wie w jaki sposób system operacyjny realizuje ochronę zasobów, segmentację pamięci oraz obsługę przerwań (K_W07)</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, dziesiętnym). Potrafi pisać programy wykorzystujące koprocessor arytmetyczny i programować z użyciem procedur. Potrafi wyjaśnić na czym polegają różne tryby adresowania, segmentacja i stronicowanie pamięci. Potrafi wyjaśnić różnicę między przerwaniem sprzętowymi a programowymi. Efekty przedmiotowe: P_U1: umie napisać prosty program w języku niskiego poziomu wykorzystujący różne tryby adresowania; potrafi używać procedur i zaimplementować rekurencję w języku assembler (K_U08) P_U2: potrafi wyjaśnić w jaki sposób działają mechanizmy obsługi przerwań, ochrony zasobów oraz pamięci wirtualnej w systemach operacyjnych (K_U08)</p>
	<p>Kompetencje społeczne (postawy)</p> <p>P_K1: Jest gotów do poszerzania swojej wiedzy w zakresie projektowania i działania systemów komputerowych (K_K01) P_K2: Jest gotów do formułowania opinii na temat ograniczeń działania systemów komputerowych (K_K03)</p>
<p>Kontakt</p> <p>piotr.szuca@ug.edu.pl</p>	