


**KAPITAŁ LUDZKI**  
 NARODOWA STRATEGIA SPÓJNOŚCI

 Projekt współfinansowany przez  
 Unię Europejską w ramach  
 Europejskiego Funduszu  
 Społecznego

**UNIA EUROPEJSKA**  
 EUROPEJSKI  
 FUNDUSZ SPOŁECZNY


<b>Nazwa przedmiotu</b>		<b>Kod ECTS</b>	
Architektura systemów komputerowych (P)		11.3.2184	
<b>Nazwa jednostki prowadzącej przedmiot</b>			
Instytut Informatyki			
<b>Studia</b>			
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>pierwszego stopnia</b>
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>			
mgr Łukasz Mielewczyk			
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>	
<b>Formy zajęć</b>		2	
Wykład, Ćw. laboratoryjne		Udział w zajęciach dydaktycznych, objętych planem studiów: 30h	
<b>Sposób realizacji zajęć</b>		Praca własna studenta: 20h	
zajęcia w sali dydaktycznej		RAZEM: 50h	
<b>Liczba godzin</b>			
Ćw. laboratoryjne: 15 godz., Wykład: 15 godz.			
<b>Termin realizacji przedmiotu</b>			
2023/2024 zimowy			
<b>Status przedmiotu</b>		<b>Język wykładowy</b>	
fakultatywny (do wyboru)		polski	
<b>Metody dydaktyczne</b>		<b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>	
<ul style="list-style-type: none"> <li>- Metoda projektów (projekt badawczy, wdrożeniowy, praktyczny)</li> <li>- Rozwiązywanie zadań</li> <li>- Wykład problemowy</li> <li>- Wykład z prezentacją multimedialną</li> </ul>		<b>Sposób zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Zaliczenie (zał)</li> </ul>	
		<b>Formy zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru</li> <li>- kolokwium</li> </ul>	
		<b>Podstawowe kryteria oceny</b>	
		O zaliczeniu ćwiczeń decyduje liczba punktów otrzymanych z kolokwium.	
<b>Sposób weryfikacji założonych efektów uczenia się</b>			

Xzakładany efekt kształcenia	egzamin	kolokwium	projekt	sprawdzian	referat	raport	aktywność w dyskusji	obserwacja postawy studenta
Wiedza								
K_W07		X						
K_W08		X						
P_W1		X						
P_W2		X						
P_W3		X						
Umiejętności								
K_U08								X
P_U1								X
P_U2								X
Kompetencje								
K_K01								X
K_K03								X
P_K1								X
P_K2								X

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

Matematyka dyskretna, Języki programowania

**B. Wymagania wstępne**

Umiejętność programowania.

Znajomość aparatu matematycznego na poziomie wykładu z Matematyki dyskretnej.

**Cele kształcenia**

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

**Treści programowe**

Kodowanie informacji liczbowych i tekstowych.

Pozycyjny zapis liczb. System dziesiętny, ósemkowy, szesnastkowy, dwójkowy.

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 ochron. Wielozadaniowość.

Przerwania programowe, sprzętowe i wyjątki

**Wykaz literatury**

- J. Biernak, "Metody i układy arytmetyki komputerowej.", Państwowe Wydawnictwo Naukowe, Warszawa 2001.
- B.S. Chalk, "Organizacja i architektura komputerów", Wydawnictwa Naukowo-Techniczne, Warszawa 1998.
- R. Hyde, "Assembler Sztuka programowania, tłumaczenie: P. Szeremiot, Helion, Gliwice 2011.
- S. Kruk, "Assembler. Podręcznik użytkownika", Wydawnictwo Mikom, Warszawa 1999.
- S. Kruk, "Procesor Pentium.", Wydawnictwo PjL, Warszawa 1998.
- D.W. Lewis, "Między assemblerem a językiem C.", Wydawnictwo RM, Warszawa 2004.
- P. Metzgerl A. Jełowicki, "Anatomia PC", Helion, Gliwice 1999.
- G. Michalek, "Assembler. Miniprzewodnik", Wydawnictwo Intersoftland, Warszawa 2001.
- A. Skorupski, "Podstawy budowy i działania komputerów", Wydawnictwa Komunikacji i Łączności, Warszawa 2004.

**Kierunkowe efekty uczenia się**

K\_W07: zna najważniejsze elementy architektury systemów

**Wiedza**

Student wie jak wygląda budowa (z punktu widzenia programisty) procesora

<p>komputerowych oraz zasady działania systemów operacyjnych ze szczególnym uwzględnieniem współbieżności, szeregowania zadań i zarządzania pamięcią oraz procesami</p> <p>K_W08: ma wiedzę na temat technologii sieciowych, w tym podstawowych protokołów komunikacyjnych, bezpieczeństwa i budowy aplikacji sieciowych</p> <p>K_U08: korzysta z zaawansowanych funkcjonalności systemów operacyjnych, w szczególności związanych z aspektami sieciowymi</p> <p>K_K01: zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się</p> <p>K_K03: potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych</p>	<p>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 bloki funkcjonalne komputera PC. Zna podstawowe grupy instrukcji procesora (adresowanie, instrukcje arytmetyczno-logiczne) oraz elementy techniki cyfrowej.</p> <p>Efekty przedmiotowe:</p> <p>P_W1: zna podstawowe polecenia języka assembler (adresowanie, instrukcje arytmetyczno-logiczne) (K_W07, K_W08)</p> <p>P_W2: zna konwencje przekazywania argumentów do procedur, sposoby przechowywania zmiennych globalnych i lokalnych w programach (ABI) (K_W07, K_W08)</p> <p>P_W3: wie w jaki sposób system operacyjny realizuje ochronę zasobów, segmentację pamięci oraz obsługę przerwań (K_W07, K_W08)</p>
	<p><b>Umiejętności</b></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 przerwaniami sprzętowymi a programowymi.</p> <p>Efekty przedmiotowe:</p> <p>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> <p>P_U2: potrafi wyjaśnić w jaki sposób działają mechanizmy obsługi przerwań, ochrony zasobów oraz pamięci wirtualnej w systemach operacyjnych (K_U08)</p>
	<p><b>Kompetencje społeczne (postawy)</b></p> <p>Efekty przedmiotowe:</p> <p>P_K1: Jest gotów do poszerzania swojej wiedzy w zakresie projektowania działania systemów komputerowych (K_K01)</p> <p>P_K2: Jest gotów do formułowania opinii na temat ograniczeń działania systemów komputerowych (K_K03)</p>
<p><b>Kontakt</b></p> <p>lukasz.mielewczyk@ug.edu.pl</p>	