

# KARTA PRZEDMIOTU

## 1. Informacje ogólne

<b>Nazwa przedmiotu i kod (wg planu studiów):</b>	Programowanie niskopoziomowe, C1
<b>Nazwa przedmiotu (j. ang.):</b>	Low Level Programming
<b>Kierunek studiów:</b>	Informatyka
<b>Specjalność/specjalizacja:</b>	Wszystkie specjalności
<b>Poziom kształcenia:</b>	studia I stopnia
<b>Profil kształcenia:</b>	praktyczny (P)
<b>Forma studiów:</b>	studia stacjonarne / studia niestacjonarne
<b>Obszar kształcenia:</b>	nauki techniczne (wg wykazu)
<b>Dziedzina:</b>	nauki techniczne (wg wykazu)
<b>Dyscyplina nauki:</b>	(wg wykazu)
<b>Koordinator przedmiotu:</b>	Prof. Dr hab. Wiesław Wajs

## 2. Ogólna charakterystyka przedmiotu

<b>Przynależność do modułu:</b>	kierunkowego
<b>Status przedmiotu:</b>	obowiązkowy
<b>Język wykładowy:</b>	polski
<b>Rok studiów, semestr:</b>	I, 1
<b>Forma i wymiar zajęć według planu studiów:</b>	stacjonarne - wykład 30 h, ćw. laboratoryjne 30 h niestacjonarne - wykład 15 h, laboratoryjne 15 h
<b>Interesariusze i instytucje partnerskie (nieobowiązkowe)</b>	
<b>Wymagania wstępne / Przedmioty wprowadzające:</b>	Dobra umiejętność posługiwania się komputerem brak

### 3. Bilans punktów ECTS

Całkowita liczba punktów ECTS (wg planu studiów; 1 punkt =25-30 godzin pracy studenta, w tym praca na zajęciach i poza zajęciami):	3 (A + B)	stacjonarne	Niestacjonarne
		<b>A. Liczba godzin wymagających bezpośredniego udziału nauczyciela (kontaktowych, w czasie rzeczywistym, w tym testy, egzaminy etc) z podziałem na typy zajęć oraz całkowita liczba punktów ECTS osiąganych na tych zajęciach</b>	obecność na wykładach obecność na ćwiczeniach laboratoryjnych udział w konsultacjach
	<b>W sumie:</b> ECTS	62 2	40 1,5
<b>B. Poszczególne typy zadań do samokształcenia studenta (niewymagających bezpośredniego udziału nauczyciela) wraz z planowaną średnią liczbą godzin na każde i sumaryczną liczbą ECTS (np. praca w bibliotece, w sieci, na platformie e-learningowej, w laboratorium, praca nad projektem końcowym, przygotowanie ogólne; suma poszczególnych godzin powinna zgadzać się z liczbą ogólną)</b>	przygotowanie ogólne przygotowanie do ćwiczeń laboratoryjnych wykonanie sprawozdań przygotowanie do kolokwium zaliczeniowego praca w bibliotece praca w sieci	2 10 5 5 4 4	5 10 5 5 5 10
	<b>w sumie:</b> ECTS	30 1	40 1,5
<b>C. Liczba godzin praktycznych/laboratoryjnych w ramach przedmiotu oraz związana z tym liczba punktów ECTS (ta liczba nie musi być powiązana z liczbą godzin kontaktowych, niektóre zajęcia praktyczne/laboratoryjne mogą odbywać się bez udziału nauczyciela):</b>	udział w ćwiczeniach laboratoryjnych praca praktyczna samodzielna	30 30	15 45
	<b>w sumie:</b> ECTS	60 2	60 2

### 4. Opis przedmiotu

<b>Cel przedmiotu:</b> Celem przedmiotu jest nabycie praktycznych umiejętności zarządzania pracą procesora w niskopoziomym języku assembler.
<b>Metody dydaktyczne:</b> wykład - pokaz, laboratorium - zadania problemowe
<b>Treści kształcenia (w rozbiciu na formę zajęć (jeśli są różne formy) i najlepiej w punktach):</b> <b>Wykłady:</b> <ol style="list-style-type: none"> <li>1. Wprowadzenie do środowiska niskopoziomowego, kompilatory, typy programów, architektura komputera, systemy liczbowe,</li> <li>2. Programy typu COM, Rejestry procesora, debugowanie kodu programu, <ol style="list-style-type: none"> <li>1. Mikroprocesor 8086 i kontroler 8288.</li> </ol> </li> <li>3. Przerwania,</li> <li>4. Rozkazy arytmetyczno logiczne,</li> <li>5. Stos, przesunięcia bitowe,</li> <li>6. Rozkazy sterujące bezwarunkowe, etykiety i procedury,</li> <li>7. Instrukcje warunkowe,</li> <li>8. Pętle,</li> <li>9. Tablice,</li> <li>10. Argumenty funkcji,</li> <li>11. Dostęp do plików na dysku,</li> <li>12. Struktura programu typu EXE.</li> </ol>

13. Protokół ModBus.
14. Analiza przykładowych programów napisanych w języku assembler.
15. Asemlacja warunkowa, podsumowanie.

**Ćwiczenia laboratoryjne:**

1. Instalacja narzędzi, kompilacja programu typu COM.
2. Rejestru procesora,
3. Przerwania,
4. Stos,
5. Rozkazy arytmetyczno-logiczne,
6. Rozkazy bezwarunkowe, definiowanie procedur,
7. Rozkazy warunkowe,
8. Pętle,
9. Tablice,
10. Argumenty funkcji,
11. Asemlacja warunkowa,
12. Operacje na plikach
13. Programy typu EXE

**5. Efekty kształcenia i sposoby weryfikacji**

Efekty kształcenia				
Efekt przedmiotu	Student, który zaliczył przedmiot (spełnił minimum wymagań)			Efekt kierunkowy
C1_K_W01 C1_K_W02	<p><b>Wiedza:</b>            Student zna zasadę wykonywania instrukcji kodu źródłowego języka assembler.            Student zna architekturę oraz listę rozkazów procesora 8086.</p>			K_W06 K_W16
C1_K_U01 C1_K_U02 C1_K_U03	<p><b>Umiejętności</b>            Student potrafi tworzyć kod źródłowy oraz kompilować, konsolidować i debugować program w języku assembler,            Student umie wykorzystać podstawowe instrukcje języka assembler.            Student potrafi wywoływać przerwania wg opisu dokumentacji oraz umie wykorzystać stos.</p>			K_U01 K_U17 K_U13 K_U03
C1_K_W01	<p><b>Kompetencje społeczne</b>            Student rozumie potrzebę wykorzystania poznanej na zajęciach wiedzy i umiejętności do programowania układów mikroprocesorowych.</p>			K_K01
Sposoby weryfikacji efektów kształcenia:				
Lp.	Efekt przedmiotu	Sposób weryfikacji	Ocena formująca poniżej	Ocena końcowa
1	C1_K_W01	Zadania problemowe na zajęciach laboratoryjnych, kolokwium	Praktyczny sprawdzian wiadomości, ocena z sprawozdania	Egzamin
2	C1_K_W02			
3	C1_K_U01	Zadania problemowe na zajęciach laboratoryjnych, kolokwium	Praktyczny sprawdzian umiejętności, ocena z sprawozdania	
4	C1_K_U02			

5	C1_K_U03			
6	C1_K_K01	Wykład	Sprawdzian ustny	
<b>Kryteria oceny</b> (oceny 3,0 powinny być równoważne z efektami kształcenia, choć mogą być bardziej szczegółowo opisane):				
<b>w zakresie wiedzy</b>				<b>Efekt kształcenia</b>
Na ocenę 3,0	Student zna zasadę wykonywania instrukcji kodu źródłowego języka assembler.		C1_K_W01	
Na ocenę 5,0	Student zna zasadę wykonywania instrukcji kodu źródłowego języka assembler, rozumie zasadę działania koprocessora.			
Na ocenę 3,0	Student zna architekturę oraz listę rozkazów procesora 8086.		C1_K_W02	
Na ocenę 5,0	Student zna architekturę oraz listę rozkazów procesora 8086 oraz rozumie zasady działania elementarnych układów cyfrowych mikroprocesora.			
<b>w zakresie umiejętności</b>				
Na ocenę 3,0	Student potrafi tworzyć kod źródłowy oraz kompilować, konsolidować i debugować program w języku assembler (plik typu COM).		C1_K_U01	
Na ocenę 5,0	Student potrafi tworzyć kod źródłowy oraz kompilować, konsolidować i debugować program w języku assembler (pliki typu COM i EXE). Zna struktury pliku COM i EXE.			
Na ocenę 3,0	Student umie wykorzystać podstawowe instrukcje języka assembler. Potrafi używać instrukcji pętli oraz instrukcji skoku warunkowego.		C1_K_U02	
Na ocenę 5,0	Student zna instrukcje języka assembler. Potrafi używać instrukcji pętli, skoku warunkowego, tworzyć procedury.			
Na ocenę 3,0	Student potrafi wywoływać proste przerwania wg opisu dokumentacji oraz umie wykorzystać stos.		C1_K_U03	
Na ocenę 5,0	Student potrafi wywoływać rozbudowane przerwania (dysku, karty graficznej) wg opisu dokumentacji oraz umie wykorzystać stos do przekazywania argumentów do procedury.			
<b>w zakresie kompetencji społecznych</b>				
Na ocenę 3,0	Student rozumie potrzebę wykorzystania poznanej na zajęciach wiedzy i umiejętności do programowania prostych układów mikroprocesorowych.		C1_K_K01	
Na ocenę 5,0	Student rozumie potrzebę wykorzystania poznanej na zajęciach wiedzy i umiejętności do programowania zaawansowanych układów mikroprocesorowych.			
<b>Kryteria oceny końcowej</b>				
kolokwia: 50 % samodzielne wykonanie ćwiczeń laboratoryjnych: 20%, aktywność za zajęciach: 15%, ocena ze sprawozdania: 15%,				
<b>Zalecana literatura</b>				
<b>Literatura podstawowa:</b>		1. Vlad Pirogow, Asembler. Podręcznik programisty, Helion, 2005. 2. Stanisław Kruk, Asembler Wykłady i ćwiczenia, Wydawnictwo Naukowe PWSN, 2009. 3. Randall Hyde, Asembler Sztuka programowania Wydanie 2, Helion, 2010.		

<b>Literatura uzupełniająca:</b>	<ol style="list-style-type: none"> <li>1. Eugeniusz Wróbel, Praktyczny kurs asemblera, Helion, 2004.</li> <li>2. Kip R. Irvine, Asembler dla procesorów Intel. Vademecum profesjonalisty, Helion, 2003.</li> <li>3. Gary Syck, Turbo Assembler Biblia Użytkownika, Oficyna Wydawnicza LT&amp;P, 1994.</li> </ol>
----------------------------------	--

**Informacje dodatkowe:**

<b>Dodatkowe obowiązki prowadzącego wraz z szacowaną całkowitą liczbą godzin:</b>
Konsultacje – 20 godzin
Przygotowanie wykładów, zajęć laboratoryjnych - 15
Przygotowanie i poprawa egzaminu – 5 godzin
W sumie: 40 godzin