

## **PROGRAMOWANIE SYSTEMOWE**

**Kod przedmiotu:** PS2

**Rodzaj przedmiotu:** kierunkowy, obowiązkowy

**Specjalność:** \_\_\_\_\_

**Wydział:** Informatyki

**Kierunek:** Informatyka

**Poziom studiów:** drugiego stopnia – VII poziom PRK

**Profil studiów:** praktyczny

**Forma studiów:** stacjonarna/niestacjonarna

**Rok:** 2

**Semestr:** 3

**Formy zajęć i liczba godzin:**

**Forma stacjonarna**

    wykłady – 15

    laboratorium - 20

**Forma niestacjonarna**

    wykłady – 10

    laboratorium - 14

**Zajęcia prowadzone są w języku polskim.**

**Liczba punktów ECTS:** 2

**Osoby prowadzące:**

    wykład:

    laboratorium:

### **1. Założenia i cele przedmiotu:**

Celem przedmiotu jest ukształtowanie wiedzy na temat metod oraz środków programowania niskopoziomowego oraz wykorzystującego narzędzia oferowane przez systemy operacyjne. Zakłada się wykształcenie umiejętności programowania z wykorzystania narzędzi niskiego poziomu, opanowanie umiejętności programowania operacji systemowych, wyrobienie umiejętności łączenia kodu niskopoziomowego z kodem języków wysokiego poziomu, wykształcenie umiejętności odpowiedniego doboru wykorzystywanych metod programowania. Istotnym elementem zajęć jest wykorzystanie mechanizmów programowania oferowanych przez wiodące systemy operacyjne, umownie zaliczanych do kategorii programowania w powłoce systemu operacyjnego.

### **2. Określenie przedmiotów wprowadzających wraz z wymaganiami wstępnymi:**

Przedmioty wprowadzające to: Programowanie obiektowe.

### 3. Opis form zajęć

#### a) Wykłady

- **Treści programowe:**
  - Programowanie niskopoziomowe, metody, narzędzia, środowiska.
  - Tryby adresacji, operacje na pamięci operacyjnej, operacje arytmetyczne, bitowe.
  - Zasady konstruowania programów, instrukcje, podprogramy.
  - Wykorzystanie przerw, przerwania sprzętowe i programowe, funkcje systemowe.
  - Elementy niskopoziomowe w językach wysokiego poziomu.
  - Łączenie kodu niskopoziomowego z kodem języków wysokiego poziomu.
  - Programowanie operacji systemowych.
  - Programowanie w środowisku powłoki systemów operacyjnych.
- **Metody dydaktyczne:**
  - Wykład prowadzony metodą tradycyjną z wykorzystaniem rzutnika multimedialnego, z wykorzystaniem materiałów udostępnianych studentom w postaci elektronicznej.
- **Forma i warunki zaliczenia:**
  - Pozytywna ocena testu zaliczeniowego realizowanego na ostatnim wykładzie w semestrze.
- **Wykaz literatury podstawowej:**
  1. Randall Hyde, Profesjonalne programowanie. Część 2. Myśl niskopoziomowo, pisz wysokopoziomowo, Gliwice, Helion.
  2. Randall Hyde, Asembler. Sztuka programowania, Gliwice, Helion.
  3. Farbaniec D.: Asembler. Programowanie. Gliwice: HELION, cop. 2019.
  4. Schwichtenberg Holger, Windows PowerShell. Podstawy, Helion.
  5. Carl Albing, JP Vossen, Cameron Newham, Bash. Receptury, Helion.
  6. Gynvael Coldwind, Mateusz Jurczyk. Praktyczna inżynieria wsteczna, PWN
- **Wykaz literatury uzupełniającej:**
  1. Randall Hyde, Profesjonalne programowanie. Część 1. Zrozumieć komputer, Gliwice, Helion.
  2. Vlad Pirogow, Asembler. Podręcznik programisty, Gliwice, Helion.

#### b) Laboratorium

- **Treści programowe:**
  - Instalacja i konfiguracja środowiska programistycznego.
  - Ćwiczenia w wykorzystaniu języka symbolicznego.
  - Języki symboliczne a operacje systemowe.
  - Programowanie niskopoziomowe w językach wysokiego poziomu.
  - Języki wysokiego poziomu a operacje systemowe.
  - Łączenie kodu niskopoziomowego z kodem języków wysokiego poziomu.
  - Programowanie w środowisku powłoki systemów operacyjnych.

- Wykorzystanie technik programowania dostępnych w środowisku wiodących systemów operacyjnych.
- **Metody dydaktyczne:**
  - Prezentacja treści i dyskusja moderowana.
  - Metoda problemowa – studium przypadku, burza mózgów.
  - Metoda laboratoryjna –ćwiczenia laboratoryjne z wykorzystaniem komputerów
- **Forma i warunki zaliczenia:**
  - Pozytywna ocena realizacji wskazanych zadań programistycznych.
  - Pozytywna ocena aktywności studenta podczas zajęć, w tym ocena biegłości w tworzeniu programów systemowych i posługiwaniu się narzędziami programistycznymi.
- **Wykaz literatury podstawowej:**
  - Joanna Drózdź, Hubert Drózdź, Skrypty w Shellu programowanie w powłoce Bash, MIKOM.
  - Ed Wilson, Windows PowerShell 3.0. Krok po kroku, Helion.
- **Wykaz literatury uzupełniającej:**
  - Jak w przypadku wykładu.

#### 4. Opis sposobu wyznaczania punktów ECTS

##### a. forma stacjonarna

Forma zajęć	Formy aktywności studenta	Średnia liczba godzin na zrealizowanie aktywności
Wykład	Kontakt z nauczycielem	15
	Czytanie wskazanej literatury	5
Laboratorium	Kontakt z nauczycielem	20
	Przygotowanie do pracy kontrolnej	10

<b>Całkowita ilość godzin aktywności studenta</b>	<b>50</b>
<b>Liczba punktów ECTS dla modułu/przedmiotu</b>	<b>2</b>

##### b. forma niestacjonarna

Forma zajęć	Formy aktywności studenta	Średnia liczba godzin na zrealizowanie aktywności
Wykład	Kontakt z nauczycielem	10
	Czytanie wskazanej literatury	10
Laboratorium	Kontakt z nauczycielem	14
	Przygotowanie do pracy kontrolnej	16

<b>Całkowita ilość godzin aktywności studenta</b>	<b>50</b>
<b>Liczba punktów ECTS dla modułu/przedmiotu</b>	<b>2</b>

#### 5. Wskaźniki sumaryczne

**a. forma stacjonarna**

- a) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich
- Liczba godzin kontaktowych – 35
  - Liczba punktów ECTS – 1,4
- b) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach o charakterze praktycznym.
- Liczba godzin kontaktowych – 20
  - Liczba punktów ECTS – 1,2

**b. forma niestacjonarna**

- a) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich
- Liczba godzin kontaktowych – 24
  - Liczba punktów ECTS – 1
- b) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach o charakterze praktycznym.
- Liczba godzin kontaktowych – 14
  - Liczba punktów ECTS – 1,2

**6. Zakładane efekty uczenia się.**

Efekt przedmiotowy (Symbol)	Efekty uczenia się dla przedmiotu	Odniesienie do kierunkowych efektów uczenia się
PS2_W1	Zna i rozumie zasady programowania w języku niskopoziomowego, rozumie zasady umieszczania elementów niskopoziomowych w językach wysokiego poziomu, łączenie kodu niskopoziomowego z kodem języków wysokiego poziomu..	IIK_W03 IIK_W05
PS2_W2	Zna i rozumie metody realizacji odwołań systemowych, wykorzystywanie funkcji systemowych, programowanie w środowisku powłoki systemu operacyjnego.	IIK_W03 IIK_W04
PS2_U1	Potrafi realizować programy niskopoziomowe, posiada umiejętność stosowania narzędzi niezbędnych do tworzenia kodu niskopoziomowego.	IIK_U11 IIK_U12
PS2_U2	Posiada umiejętność wykorzystywania elementów niskopoziomowych w kodzie języków wysokiego poziomu, potrafi ocenić użyteczność optymalizacji z wykorzystaniem programowania niskopoziomowego, potrafi wykorzystywać języki programowania powłoki systemu operacyjnego.	IIK_U07 IIK_U08
PS2_K1	Posiada świadomość konieczności doboru odpowiednich metod i narzędzi programowania, potrafi ocenić ich użyteczność i wpływ na funkcjonowanie w warunkach realnego wdrożenia.	IIK_K01 IIK_K02

**7. Odniesienie efektów uczenia się do form zajęć i sposób oceny osiągnięcia przez studenta efektów uczenia się.**

Efekt przedmiotowy (Symbol)	Forma zajęć		Sposób sprawdzenia osiągnięcia efektu
	Wykład	Laboratorium	
PS2_W1	v		Test zaliczeniowy
PS2_W2	v	v	Test zaliczeniowy
PS2_U1		v	Prace kontrolne
PS2_U2		v	Prace kontrolne
PS2_K1		v	Prace kontrolne, ocena aktywności

**8. Kryteria uznania osiągnięcia przez studenta efektów uczenia się.**

Efekt przedmiotowy (Symbol)	Efekt jest uznawany za osiągnięty, gdy student:
PS2_W1	Zalicza ponad 50% pytań w teście sprawdzającym.
PS2_W2	Poprawnie rozwiązuje zadania w czasie zajęć. Zalicza ponad 50% pytań w teście sprawdzającym.
PS2_U1	Poprawnie rozwiązuje zadania w czasie zajęć. Zalicza ponad 50% pytań w teście sprawdzającym.
PS2_U2	Osiąga ponad 50% punktów w pracach kontrolnych.
PS2_K1	Osiąga ponad 50% punktów w pracach kontrolnych.