



## Podstawy programowania Sylabus zajęć

### Informacje podstawowe

<b>Kierunek studiów</b> Informatyka	<b>Cykl dydaktyczny</b> 2023/24
<b>Specjalność</b> -	<b>Kod zajęć</b> 06INFS.31P.00129.23
<b>Jednostka organizacyjna</b> Wydział Matematyki i Informatyki	<b>Języki wykładowe</b> polski
<b>Poziom studiów</b> studia inżynierskie pierwszego stopnia	<b>Obligatoryjność</b> Obowiązkowy
<b>Forma studiów</b> studia stacjonarne	<b>Blok zajęciowy</b> Przedmioty podstawowe
<b>Profil studiów</b> profil ogólnoakademicki	
<b>Koordynator zajęć</b>	Jerzy Szymański
<b>Prowadzący zajęcia</b>	Jerzy Szymański
<b>Okres</b> Semestr 1	<b>Forma zajęć / liczba godzin / forma zaliczenia</b> • Wykład: 30, Egzamin • Ćwiczenia w salach komputerowych: 60, Zaliczenie z oceną
	<b>Liczba punktów ECTS</b> 6

### Cele kształcenia dla zajęć

Kod	Cel
C1	Zapoznanie studentów z wybranymi zagadnieniami architektury komputera oraz podstawowymi koncepcjami imperatywnych języków programowania, z przykładami w językach C/C++, Python oraz Java.
C2	Wyposażenie studentów w wiedzę i umiejętności niezbędne do implementowania standardowych algorytmów i struktur danych w tych językach programowania.

## Wymagania wstępne

Brak.

### Efekty uczenia się dla zajęć

Kod	Efekty uczenia się dla zajęć w zakresie	Efekty uczenia się dla kierunku	Metody weryfikacji osiągnięcia efektów uczenia się dla zajęć
<b>Wiedzy - Student/ka:</b>			
W1	Zna logiczną zasadę funkcjonowania komputera oraz działania programu (maszyna RAM).	INF_K3_W03, INF_K3_W04_inz, INF_K3_W06_inz	Egzamin pisemny
W2	Zna podstawowe konstrukcje programistyczne (instrukcje sterujące, wywoływanie procedur i funkcji oraz różne typy przekazywania parametrów).	INF_K3_W03, INF_K3_W04_inz	Egzamin pisemny, Projekt, Zadania programistyczne
W3	Zna podstawowe wbudowane typy danych oraz zasady tworzenia złożonych typów danych.	INF_K3_W04_inz, INF_K3_W05	Egzamin pisemny, Projekt, Zadania programistyczne
W4	Zna zasady dynamicznego zarządzania pamięcią.	INF_K3_W06_inz	Egzamin pisemny, Projekt, Zadania programistyczne
W5	Zna i rozumie paradygmaty programowania obiektowego (abstrakcja, enkapsulacja, dziedziczenie i polimorfizm).	INF_K3_W04_inz, INF_K3_W05	Egzamin pisemny
<b>Umiejętności - Student/ka:</b>			
U1	Potrafi stosować podstawowe konstrukcje programistyczne (instrukcje sterujące, wywoływanie procedur i funkcji oraz różne typy przekazywania parametrów).	INF_K3_U04_inz, INF_K3_U05_inz	Egzamin pisemny, Projekt, Zadania programistyczne
U2	Potrafi czytać i analizować kod "proceduralny" napisany w językach C oraz Python i rozumie istotę i efekty wykonywanych operacji.	INF_K3_U05_inz	Egzamin pisemny, Projekt, Zadania programistyczne
U3	Potrafi czytać i analizować kod obiektowy napisany w językach C++, Python oraz Java.	INF_K3_U04_inz, INF_K3_U05_inz	Projekt, Zadania programistyczne
U4	Potrafi pisać programy w stylu obiektowym z wykorzystaniem wszystkich paradygmatów programowania obiektowego.	INF_K3_U04_inz	Projekt, Zadania programistyczne

### Treści programowe dla zajęć

Lp.	Treści programowe dla zajęć	Efekty uczenia się dla zajęć	Formy zajęć
1.	Logiczna zasada działania komputera i programu (maszyna RAM). Tworzenie oprogramowania. Algorytm, procesor, program, język programowania. Kompilacja, interpretacja i konsolidacja programu. Dane i kod.	W1, W3, U2	Wykład, Ćwiczenia w salach komputerowych
2.	Sterowanie przebiegiem programu.	W1, W2, U1, U2	Wykład, Ćwiczenia w salach komputerowych

Lp.	Treści programowe dla zajęć	Efekty uczenia się dla zajęć	Formy zajęć
3.	Typ danych, typy podstawowe, reprezentacja danych. Stałe i zmienne.	W3, W4, U2	Wykład, Ćwiczenia w salach komputerowych
4.	Złożone typy danych i zarządzanie pamięcią.	W3, W4, U1, U2	Wykład, Ćwiczenia w salach komputerowych
5.	Podprogramy. Przekazywanie parametrów. Przeciążanie, funkcje biblioteczne.	W2, W4, U1, U2	Wykład, Ćwiczenia w salach komputerowych
6.	Obiekty i klasy, pola, metody. Enkapsulacja, dziedziczenie i polimorfizm, klasy abstrakcyjne, interfejsy, tworzenie obiektów oraz korzystanie z ich pól i metod.	W2, W3, W5, U1, U3	Wykład, Ćwiczenia w salach komputerowych
7.	Graficzny interfejs użytkownika.	W3, W4, W5, U1, U3, U4	Wykład, Ćwiczenia w salach komputerowych

### Informacje dodatkowe

Forma zajęć	Metody i formy prowadzenia zajęć
Wykład	Wykład z prezentacją multimedialną wybranych zagadnień
Ćwiczenia w salach komputerowych	Rozwiązywanie zadań (np.: obliczeniowych, artystycznych, praktycznych), Metoda projektu, Pokaz i obserwacja

Forma zajęć	Warunki zaliczenia zajęć
Wykład	Warunkiem przystąpienia do egzaminu jest uzyskanie zaliczenia z ćwiczeń w salach komputerowych. Na końcową ocenę składa się wynik uzyskany na egzaminie pisemnym. Skala ocen: 1. bardzo dobry (bdb; 5,0) – od 90% punktów, 2. dobry plus (db plus; 4,5) – od 80% punktów, 3. dobry (db; 4,0) – od 70% punktów, 4. dostateczny plus (dst plus; 3,5) – od 60% punktów, 5. dostateczny (dst; 3,0) – od 50% punktów, 6. niedostateczny (ndst; 2,0) – poniżej 50% punktów.
Ćwiczenia w salach komputerowych	Końcowa ocena składa się z projektu oraz zadań programistycznych. Skala ocen: 1. bardzo dobry (bdb; 5,0) – od 90% punktów, 2. dobry plus (db plus; 4,5) – od 80% punktów, 3. dobry (db; 4,0) – od 70% punktów, 4. dostateczny plus (dst plus; 3,5) – od 60% punktów, 5. dostateczny (dst; 3,0) – od 50% punktów, 6. niedostateczny (ndst; 2,0) – poniżej 50% punktów.

## Literatura

### Obowiązkowa

1. S. Prata (2016). Język C. Szkoła programowania. Wydanie 6. Helion.
2. J. Grębosz (2020). Opus magnum C++11. Programowanie w języku C++. Wydanie 2. Helion.
3. J. Bloch (2018). Java. Efektywne programowanie. Wydanie 3. Helion.
4. C. Horstmann (2022). Java. Podstawy. Wydanie 12. Helion.

### Dodatkowa

1. N. Wirth, Wprowadzenie do programowania systematycznego. WNT. Warszawa 1978

## Nakład pracy studenta i punkty ECTS

Rodzaje zajęć studenta	Średnia liczba godzin* przeznaczonych na zrealizowane rodzaje zajęć
Wykład	30
Ćwiczenia w salach komputerowych	60
Przygotowanie do zajęć	15
Przygotowanie projektu	25
Przygotowanie do egzaminu	20
<b>Łączny nakład pracy studenta</b>	<b>Liczba godzin</b> 150
<b>Liczba punktów ECTS</b>	<b>ECTS</b> 6

\* godzina (lekcyjna) oznacza 45 minut

## Efekty uczenia się dla kierunku

Kod	Treść
INF_K3_U04_inz	Absolwent/ka potrafi opracować, przeanalizować, zaprojektować klasyczne algorytmy i systemy informatyczne
INF_K3_U05_inz	Absolwent/ka potrafi pisać, uruchamiać i testować programy w wybranym środowisku programistycznym
INF_K3_W03	Absolwent/ka zna i rozumie narzędzia, technologie i urządzenia informatyczne właściwe dla wybranych obszarów zastosowań oraz podstawy ich działania
INF_K3_W04_inz	Absolwent/ka zna i rozumie zaawansowane pojęcia, konstrukcje i procesy związane z językami programowania i inżynierią programowania
INF_K3_W05	Absolwent/ka zna i rozumie zaawansowane pojęcia związane z algorytmami i strukturami danych
INF_K3_W06_inz	Absolwent/ka zna i rozumie zaawansowane pojęcia związane z informatyczną architekturą sprzętowo-programową