

PODSTAWY PROGRAMOWANIA

Kod przedmiotu: 11.3-WK-ID-PP

Forma zajęć	Liczba godzin w semestrze	Liczba godzin w tygodniu	Semestr	Forma zaliczenia	Punkty ECTS
Studia stacjonarne					5
Wykład	30	2	I	Egzamin	
Laboratorium	30	2		Zaliczenie na ocenę	

ZAKRES TEMATYCZNY PRZEDMIOTU:

Wykład

- **System komputerowy.** Sprzęt i oprogramowanie. Arytmetyka komputerów. Algorytm, język programowania, program, kompilator, sieć działań programu
- **Podstawy programowania w C++.** Paradygmat programowania strukturalnego. Struktura programu. Standardowe typy proste: całkowite, rzeczywiste, logiczny, znakowy, łańcuchowy. Deklaracje zmiennych. Instrukcja przypisania. Instrukcje wejścia-wyjścia. Funkcje: definicje, obiekty lokalne i globalne, przekazywanie parametrów przez zmienną i przez wartość, funkcje standardowe. Instrukcje warunkowe i iteracyjne. Tablice, operacje na tablicach. Przetwarzanie łańcuchów.
- **Własności algorytmów.** Asymptotyka. Złożoność obliczeniowa algorytmów. Semantyczna poprawność.
- **Zmienne dynamiczne:** definicje, procedury tworzenia i usuwania zmiennych dynamicznych, operacje określone na elementach typu wskaźnikowego
- **Pliki.** Podstawowe operacje na plikach w C++.

Laboratorium

- System Dev C++.
- Podstawowe elementy języka C++: Standardowe typy proste, wyrażenia arytmetyczne i logiczne, deklaracje zmiennych, wybrane funkcje standardowe.
- Pisanie i uruchamianie prostych programów z instrukcjami: przypisania, wejścia-wyjścia, warunkowymi.
- Pisanie i uruchamianie programów zawierających procedury i funkcje oraz programów zawierających instrukcje iteracyjne.
- Przetwarzanie tablic i łańcuchów.
- Zmienne dynamiczne i proste struktury dynamiczne.

EFEKTY KSZTAŁCENIA:

Student:

1. potrafi ułożyć i analizować algorytm zgodny ze specyfikacją,
2. potrafi interpretować i analizować przykładowe programy napisane w języku C++,
3. potrafi samodzielnie napisać program w C++ zawierający podstawowe konstrukcje języka (stałe, zmienne, tablice, instrukcje przypisania, wejścia-wyjścia, warunkowe, iteracyjne, procedury, funkcje), skompilować go, uruchomić i testować,
4. ma podstawową wiedzę na temat arytmetyki komputerów, paradygmatów programowania, algorytmów oraz ich złożoności obliczeniowej i pamięciowej,
5. rozumie potrzebę ciągłego podnoszenia swoich kwalifikacji poprzez poszerzanie swojej wiedzy i praktycznych umiejętności.