



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie i język C [S1ETI1>PiJC]

### Przedmiot

Kierunek studiów

Edukacja techniczno-informatyczna

Rok/Semestr

1/2

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

### Liczba godzin

Wykład

15

Laboratorium

30

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

### Liczba punktów ECTS

3,00

### Koordynatorzy

dr hab. inż. Małgorzata Jankowska  
malgorzata.jankowska@put.poznan.pl

### Wykładowcy

### Wymagania wstępne

Podstawowa wiedza z informatyki oraz matematyki. Umiejętność programowania w dowolnym języku wysokiego poziomu na poziomie podstawowym, algorytmizacji zadań oraz logicznego i abstrakcyjnego myślenia. Zrozumienie potrzeby tworzenia programów komputerowych w celu zwiększenia wydajności pracy, obliczeń, wizualizacji wyników oraz prezentacji informacji zgromadzonych w bazach danych.

### Cel przedmiotu

Głównym celem przedmiotu jest zapoznanie studenta z zasadami programowania w języku C/C++, zdobycie przez niego umiejętności samodzielnego tworzenia aplikacji wykorzystujących podstawowe cechy tego języka oraz algorytmizacji zadań.

### Przedmiotowe efekty uczenia się

Wiedza:

w01 znajomość pojęć i zagadnień informatycznych niezbędnych do zrozumienia filozofii programowania w językach wysokiego poziomu, w tym w języku c/c++. k1\_w01, k1\_w08

w02 znajomość i rozumie elementów języka c/c++ niezbędnych do tworzenia rozbudowanych aplikacji informatycznych w zakresie obejmowanym przez treści programowe właściwe dla kierunku

studiów. k1\_w01, k1\_w08

w03 wiedza niezbędna do samodzielnego tworzenia algorytmów implementowanych w języku c/c++.

k1\_w01, k1\_w08

Umiejętności:

u01 umiejętność tworzenia programów komputerowych z wykorzystaniem języka programowania

c/c++. k1\_u01, k1\_u02, k1\_u04, k1\_u11

u02 zdolność wykorzystania zdobytych umiejętności w zakresie tworzenia algorytmów i efektywnej

implementacji zadań obliczeniowych i problemów inżynierskich. k1\_u01, k1\_u02, k1\_u04, k1\_u11

u03 umiejętność samokształcenia w zakresie poznawania zaawansowanych elementów języka c/c++

oraz innych języków wysokiego poziomu. k1\_u01, k1\_u02, k1\_u04, k1\_u11

u04 umiejętność pozyskiwania informacji z literatury, baz danych oraz innych dostępnych źródeł

wiedzy. k1\_u01, k1\_u02, k1\_u04, k1\_u11

Kompetencje społeczne:

k01 odpowiedzialność za powierzone zadania, np. samodzielny projekt programistyczny. k1\_k01,

k1\_k03, k1\_k05

k02 rozumienie potrzeby ciągłego dokształcania się w celu podnoszenia kompetencji zawodowych,

zdobywania aktualnej wiedzy z zakresu języków programowania oraz informatyki (np. poprzez czytanie

czasopism informatycznych, uczestnictwie w kursach i studiach podyplomowych). k1\_k01, k1\_k03,

k1\_k05

k03 umiejętność pracy nad wyznaczonym zadaniem samodzielnie oraz w zespole przyjmując w nim

różne role. k1\_k01, k1\_k03, k1\_k05

k04 rozumienie znaczenia algorytmizacji zadań w procesie tworzenia efektywnych programów

komputerowych. k1\_k01, k1\_k03, k1\_k05

## Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

W01 Pisemne kolokwia obejmujące materiał wykazany w treściach programowych. Ocena końcowa uzależniona od procentowej liczby punktów:

3 50.1%-70.0%

4 70.1%-90.0%

5 od 90.1%

U01 Zadania projektowe wymagane w formie programów i oceniane pod koniec zajęć laboratoryjnych oraz na kolokwiah zaliczeniowych. Ocena końcowa uzależniona od procentowej liczby punktów:

3 50.1%-70.0%

4 70.1%-90.0%

5 od 90.1%

## Treści programowe

Tematyka modułu obejmuje zagadnienia związane z podstawami programowania w językach C i C++. W ramach treści programowych zakłada się omówienie typów danych i zmiennych, operatorów, wskaźników, tablic i struktur, wyrażeń i instrukcji języków C/C++, operacji wejścia/wyjścia na plikach oraz funkcji.

## Tematyka zajęć

Wykłady:

1. Podstawowe pojęcia i zagadnienia:

- Języki programowania (języki niskiego i wysokiego poziomu oraz ich charakterystyka).

- Historia języków programowania C/C++ oraz ich zalety.

- Standard języka C++.

- Wprowadzenie do pracy w zintegrowanym środowisku programistycznym Visual Studio.

- Systemy liczbowe - binarny i heksadecymalny system liczbowy.

- Standardy reprezentowania cyfr, liter i znaków specjalnych.

2. Typy danych i zmienne:

- Definicja i podział na typy proste (skalarne) i złożone (strukturalne).

- Charakterystyka typów prostych - całkowitoliczbowego, zmiennopozycyjnego, logicznego, znakowego.

3. Jednostki leksykalne języka C++ (słowa kluczowe, identyfikatory, literały, stałe).
4. Operatory:
  - Operatory arytmetyczne (addytywne, multiplikatywne, inkrementacji, dekrementacji).
  - Operatory logiczne (koniunkcji, alternatywy, negacji).
  - Operatory przypisania i porównania.
  - Operator warunkowy.
5. Wskaźniki, tablice i struktury:
  - Typy i zamienne wskaźnikowe - deklaracja, inicjacja i dereferencja.
  - Typ void - wskazania adresowe typu void.
  - Typy i zmienne referencyjne.
  - Tablice statyczne - deklaracja, inicjacja, dostęp do elementów tablicy.
  - Struktury - deklaracja i dostęp do elementów struktury.
  - Tablice dynamiczne.
6. Wyrażenia i instrukcje języka C++.
7. Arytmetyka wskaźników – związek między tablicami statycznymi i wskaźnikami..
8. Operacje wejścia/wyjścia na plikach.
9. Funkcje języka C++.

Laboratoria:

1. Wprowadzenie do programowania w języku C/C++. Typy danych, zmienne, stałe, jednostki leksykalne, operatory.
2. Tablice statyczne jedno- i dwuwymiarowe. Instrukcje języka C/C++:
  - Instrukcje warunkowe. Działania arytmetyczne. Wyrażenia logiczne i relacje.
  - Standardowe funkcje matematyczne.
  - Instrukcje iteracyjne.
  - Instrukcje sterujące.
  - Tablice statyczne jednowymiarowe. Generowanie liczb pseudolosowych.
  - Instrukcja switch. Operacje na plikach tekstowych.
  - Tablice statyczne dwuwymiarowe.
3. Struktury. Statyczne i dynamiczne tablice struktur.
4. Funkcje języka C/C++:
  - Jednowymiarowe tablice dynamiczne. Funkcje.
  - Dwuwymiarowe tablice dynamiczne. Funkcje.

## Metody dydaktyczne

Wykłady:

Prezentacja multimedialna ilustrowana przykładowymi programami napisanymi w języku C/C++.

Laboratoria:

Tworzenie programów komputerowych w języku C/C++.

## Literatura

Podstawowa

1. H. M. Deitel, P. J. Deitel, Arkana C++ Programowanie, Wydawnictwo RM, Warszawa 1998.
2. S. Prata, Szkoła Programowania. Język C++, Wydawnictwo Helion, Gliwice 2006.
3. A. Zalewski, Programowanie w językach C i C++ z wykorzystaniem pakietu Borland C++, Wydawnictwo Nakom, Poznań 1996.
4. J. Grębosz, Symfonia C++. Programowanie w języku C++ orientowane obiektowo, Tom 1,2,3, Oficyna Kallimach, Kraków 1999.

Uzupełniająca

1. D. E. Knuth, Sztuka programowania. Tom1 Algorytmy podstawowe, Wydawnictwa Naukowo-Techniczne, Warszawa 2002.
2. N. Wirth, Algorytmy + struktury danych = programy, Wydawnictwa Naukowo-Techniczne, Warszawa 2004.

## Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	86	3,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	51	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwii/egzaminu, wykonanie projektu)	50	2,00