

OPISY KURSÓW

- Kod kursu: **ARR2514**
- Nazwa kursu: **PROGRAMOWANIE W JĘZYKU C**
- Język wykładowy: **polski, angielski**

<i>Forma kursu</i>	<i>Wykład</i>	<i>Ćwiczenia</i>	<i>Laboratorium</i>	<i>Projekt</i>	<i>Seminarium</i>
<i>Tygodniowa liczba godzin ZZU *</i>	2		2		
<i>Semestralna liczba godzin ZZU*</i>	30		30		
<i>F o r m a zaliczenia</i>	zaliczenie		zaliczenie		
<i>Punkty ECTS</i>	3		2		
<i>Liczba godzin CNPS</i>	90		60		

- Poziom kursu (podstawowy/zaawansowany): **podstawowy**
- Wymagania wstępne: **zaliczenie kursu wprowadzającego do zagadnień informatycznych (Technologie Informacyjne)**
- Imię, nazwisko i tytuł/ stopień prowadzącego:
dr hab. inż. Kazimierz Wilkosz, prof. nadzw.
- Imiona i nazwiska oraz tytuły/stopnie członków zespołu dydaktycznego:
dr inż. Kazimierz Herlender,
dr inż. Anna Kisiel,
dr inż. Mieczysław Kozak
dr inż. Robert Lis,
dr inż. Mirosław Łabuzek,
dr inż. Robert Łukomski,
dr inż. Jerzy Piotrowicz,
dr inż. Jarosław Szymańda,
dr inż. Leszek Woźny,
dr inż. Ryszard Zacirka.
- Rok: **1. Semestr: 2**
- Typ kursu (obowiązkowy/wybieralny): **obowiązkowy**
- Cele zajęć (efekty kształcenia):
 - **zaznajomienie i nabranie biegłości w posługiwaniu się zasadami podejścia strukturalnego do tworzenia algorytmów,**
 - **umiejętność pisania programów w języku C,**
 - **zaznajomienie z zasadniczymi ideami programowania obiektowego z wykorzystaniem języka C++,**
 - **pogłębienie umiejętności wykorzystania komputerów.**
- Forma nauczania (tradycyjna/zdalna): **tradycyjna**

- Krótki opis zawartości całego kursu:
Charakterystyka zintegrowanego środowiska programowania z wykorzystaniem języka C. Struktura programu w języku C oraz C++. Typy danych. Stałe. Zmienne. Operacje wejścia/wyjścia. Operatory. Wyrażenia. Instrukcje. Funkcje. Wskaźniki. Tablice. Struktury danych. Pliki. Operacje na łańcuchach znaków. Porównanie języka C z językiem C++. Wprowadzenie do programowania obiektowego: obiekty, klasy, konstruktory, destruktory, dziedziczenie. Podstawy tworzenia algorytmów.
- Wykład (podać z dokładnością do 2 godzin):

<i>Zawartość tematyczna poszczególnych godzin wykładowych</i>	<i>Liczba godzin</i>
1. Wprowadzenie. Algorytm. Schematy blokowe. Idea programowania strukturalnego.	2
2. Struktura programów w C. Identyfikator typy danych (typy fundamentalne: całkowite, rzeczywiste, znakowe, logiczny), deklaracja i inicjalizacja zmiennych, definiowanie stałych. Komunikacja poprzez konsolę. Operatory: arytmetyczne, logiczne, inkrementacji, dekrementacji, przypisania. Obliczanie wartości wyrażeń.	2
3. Struktury sterowania obliczeniami: rozgałęzienia i skoki, pętle pojedyncze i zagnieżdżone. Instrukcje proste i złożone; instrukcje warunkowe, wyrażenia warunkowe, instrukcje iteracyjne.	2
4. Preprocesor: dyrektywy, makrodefinicje.	2
5. Funkcje: budowa funkcji, argumenty funkcji, wynik wykonania funkcji, definicje i deklaracje globalne, argumenty funkcji main, rekurencja.	2
6. Tablice (tablice jedno i wielowymiarowe), łańcuchy znaków.	2
7. Wskaźniki. Pamięć dynamiczna.	2
8. Kolokwium.	2
9. Struktury danych, unie: deklaracja struktury, definiowanie zmiennej strukturalnej, tablice struktur, wskaźniki a struktury danych.	2
10. Operacje na plikach: otwieranie, zamykanie plików, czytanie i zapisywanie do plików.	2
11. Operacje na łańcuchach znaków. Formatowanie w operacjach wejście/wyjście. Binarne wejście/wyjście.	2
12. Idea programowania obiektowego. Struktura programów w C++. Obiekty. Klasy.	2
13. Konstruktory, destruktory.	2
14. Relacje między klasami: zaprzyjaźnienie. Dziedziczenie.	2
15. Kolokwium zaliczeniowe.	2

- Ćwiczenia - zawartość tematyczna:
- Seminarium - zawartość tematyczna:
- Laboratorium - zawartość tematyczna:
 1. Wprowadzenie do laboratorium. Schematy blokowe dla prostych algorytmów.
 2. Schematy blokowe dla bardziej złożonych algorytmów.
 3. Pisanie, kompilacja i uruchamianie prostych programów - wyświetlanie liczb i napisów na ekranie monitora.
 4. Pisanie programów z wykorzystaniem rozgałęzień, skoków i pętli.

5. Wykorzystanie dyrektywy i makrodefinicji.
 6. Programowanie z wykorzystaniem funkcji.
 7. Programowanie operacji na tablicach.
 8. Wprowadzenie wskaźników do programów.
 9. Pisanie programów z wykorzystaniem struktur oraz unii.
 10. Tworzenie programów przewidujących wczytywanie danych wejściowych z plików i zapisywanie wyników do plików.
 11. Programowanie z wykorzystaniem obiektów oraz klas.
 12. Wykorzystanie konstruktorów i destruktorów.
 13. Stosowanie klas. Zaprzyjaźnienie. Dziedziczenie.
- Projekt - zawartość tematyczna:
 - Literatura podstawowa:
 1. Wirth N., Algorytmy + struktury danych = programy. Warszawa, WNT 2001.
 2. Kernighan B. W., Ritchie D. M., Język ANSI C, Warszawa, WNT 2003.
 3. Sexton C., Język C to proste, Warszawa, Wyd. RM 2001.
 4. Grębosz J., Symfonia C++, Kraków, Kallimach 2000.
 - Literatura uzupełniająca:
 1. Kubiak M. J., Programuję w językach Turbo Pascal i C/C++: programowanie strukturalne z elementami programowania obiektowego, Warszawa, Mikom 2001.
 2. Stec K., Wybrane elementy języka C, Gliwice, Wyd. Pol. Śląskiej 2001.
 3. Sexton C., Programowanie w C++ - to proste, Warszawa, RM 2001.
 - Warunki zaliczenia:
Wykład: pozytywna ocena z kolokwium zaliczeniowego oraz prac własnych
Laboratorium: pozytywne oceny z wykonanych ćwiczeń

* - w zależności od systemu studiów