**Nazwa przedmiotu:**

Zasady programowania strukturalnego II -z

**Koordynator przedmiotu:**

prof. nzw. dr hab. inż. Barbara Putz

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Mechatronika

**Grupa przedmiotów:**

Obowiązkowe

**Kod przedmiotu:**

ZAPIInst

**Semestr nominalny:**

3 / rok ak. 2012/2013

**Liczba punktów ECTS:**

3

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

Punkty ECTS:
- projektowanie w pracowni komputerowej 12,
- studia literaturowe 8,
- opracowanie projektu poza pracownią 40,
- opracowanie dokumentacji projektu 15.
RAZEM 75 godz. = 3 ECTS

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

Punkty ECTS:
- projektowanie w pracowni komputerowej 12.
Razem 12 godz. = 1 ECTS

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

Punkty ECTS:
- obecność na zajęciach 12,
- projektowania poza pracownią 40,
- dokumentowanie projektu 15.
RAZEM 67 godz. = 2 ECTS

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład:  | 0h |
| Ćwiczenia:  | 0h |
| Laboratorium:  | 0h |
| Projekt:  | 180h |
| Lekcje komputerowe:  | 0h |

**Wymagania wstępne:**

Zaliczony (wskazane) przedmiot Zasady programowania strukturalnego I lub ukończony kurs podstaw programowania strukturalnego.

**Limit liczby studentów:**

max. 15 studentów

**Cel przedmiotu:**

Nauka logicznego, algorytmicznego myślenia przy rozwiązywaniu problemów inżynierskich.
Zdobycie umiejętności tworzenia złożonych programów o strukturze modułowej.

**Treści kształcenia:**

Indywidualne projekty są realizowane w języku C/C++.

W tworzonej aplikacji wymagane jest:
- wprowadzanie i/lub zapisywanie danych do programu z wykorzystaniem typu plikowego,
- struktura danych do projektu musi też wykorzystywać tablice dynamiczne lub listy,
- aplikacja musi mieć budowę modułową i funkcje programu muszą być umieszczane w utworzonych dla niej bibliotekach.
Aplikacja musi być udokumentowana i testowana.

**Metody oceny:**

Ocena zaliczająca przedmiot jest ustalana na podstawie ocen cząstkowych z wagami z etapów wykonywania indywidualnego projektu: algorytmu projektu, prototypu i ostatecznie wykonanej aplikacji.

**Egzamin:**

nie

**Literatura:**

1) B. Putz, P. Wnuk: Informatyka 2 - Programowanie. OKNO PW, Warszawa 2002.
2) P. Wnuk, B. Putz: Informatyka 2 - Programowanie. Wersja w języku C/C++. OKNO PW, Warszawa 2005.
3) S. Prata: Język C. Szkoła programowania. Robomatic 2001.
4) N. Wirth: Algorytmy+struktury danych=programy. WNT 2002.
5) P.Wróblewski: Algorytmy, struktury danych i techniki programowania. Helion 2003.
6)R. Neapolitan, Kumarss Naimipour: Podstawy algorytmów z przykładami w C++. Helion 2004.

**Witryna www przedmiotu:**

http://iair.mchtr.pw.edu.pl/studenci/

**Uwagi:**

Przedmiot daje gruntowne podstawy do programowania, umożliwiając tworzenie własnych aplikacji do rozwiązywania różnych problemów technicznych.

## Efekty przedmiotowe

### Profil ogólnoakademicki - wiedza

**Efekt ZAPIInst\_W01:**

Posiada praktyczną wiedzę w zakresie algorytmizacji problemów technicznych

Weryfikacja:

Zaliczenie

**Powiązane efekty kierunkowe:** K\_W04, K\_W05

**Powiązane efekty obszarowe:** T1A\_W02, T1A\_W02, T1A\_W04

**Efekt ZAPIInst\_W02:**

Ma wiedzę w zakresie programowania strukturalnego

Weryfikacja:

Zaliczenie

**Powiązane efekty kierunkowe:** K\_W04, K\_W05

**Powiązane efekty obszarowe:** T1A\_W02, T1A\_W02, T1A\_W04

### Profil ogólnoakademicki - umiejętności

**Efekt ZAPIInst\_U01:**

Ma praktyczną umiejętność tworzenia złożonych programów

Weryfikacja:

Zaliczenie

**Powiązane efekty kierunkowe:** K\_U01, K\_U03

**Powiązane efekty obszarowe:** T1A\_U01, T1A\_U04

### Profil ogólnoakademicki - kompetencje społeczne

**Efekt ZAPIInst\_K01:**

Rozumie potrzebę ciągłego kształcenia

Weryfikacja:

Zaliczenie

**Powiązane efekty kierunkowe:** K\_K01

**Powiązane efekty obszarowe:** T1A\_K01