**Nazwa przedmiotu:**

Podstawy programowania

**Koordynator przedmiotu:**

mgr inż. Marcin Ścibisz

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Zarządzanie

**Grupa przedmiotów:**

obowiązkowe

**Kod przedmiotu:**

PODPR

**Semestr nominalny:**

3 / rok ak. 2013/2014

**Liczba punktów ECTS:**

4

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

100h (4 ECTS):
14h (wykład) + 14h (laboratorium) + 1h (kons. grupowe) + 1h (kons. indywidualne) + 14x2h (przygotowanie do ćwiczeń laboratoryjnych) + 10h (przygotowanie do kolokwium – wykład) + 16h (przygotowanie do kolokwium - laboratorium) + 16h (przygotowanie aplikacji)

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

1,2 ECTS:
14h (wykład) + 14h (laboratorium) + 1h (kons. grupowe) + 1h (kons. indywidualne) = 30h

**Język prowadzenia zajęć:**

polski

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

3 ECTS:
14h (laboratorium) + 14x2h (przygotowanie do ćwiczeń laboratoryj-nych) + 16h (przygotowanie do kolokwium - laboratorium) + 16h (projekt i implementacja aplikacji) = 74h

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

|  |  |
| --- | --- |
| Wykład:  | 210h |
| Ćwiczenia:  | 0h |
| Laboratorium:  | 210h |
| Projekt:  | 0h |
| Lekcje komputerowe:  | 0h |

**Wymagania wstępne:**

umiejętność obsługi komputera, posługiwanie się pocztą elektroniczną oraz edytorem tekstu

**Limit liczby studentów:**

od 15 osób do limitu miejsc w sali (wykład); od 10 do 30 (zajęcia laboratoryjne / projekty komputerowe)

**Cel przedmiotu:**

Celem przedmiotu jest, aby po jego zaliczeniu student:
- posiadał podstawową wiedzę z zakresu programowania w języku wysokiego poziomu z położeniem nacisku na programowanie zorientowane obiektowo,
- potrafił implementować proste algorytmy oraz typowe struktury danych,
- potrafił pisać oraz uruchamiać proste programy o rozmiarze rzędu 100-150 wierszy kodu źródłowego z wykorzystaniem zintegrowanego środowiska programistycznego.

**Treści kształcenia:**

Wykład: 1) Struktura przedmiotu, podstawowa terminologia, wprowadzenie do programowania 2) Wprowadzenie do języka Java. 3) Proste typy danych. 4.)Wyrażenie i operatory. 5) Instrukcje sterujące. 6) Złożone typy danych – łańcuchy. 7) Złożone typy danych – tablice jednowymiarowe. 8) Złożone typy danych – tablice wielowymiarowe. 9) Programowanie obiektowe w języku Java. 10) Tworzenie własnych klas. 11) Dziedziczenie. 12) Interfejsy i klasy wewnętrzne. 13) Wyjątki i obsługa błędów. 14) Strumienie i pliki. 15) Strumienie obiektów.
Laboratorium: Wykonanie w laboratorium ćwiczeń z wykorzystaniem zintegrowanego środowiska programistycznego w zakresie: 1) Proste typy danych. 2) Wyrażenia i operatory. 3) Instrukcje sterujące warunkowe. 4) Instrukcje sterujące iteracyjne. 5) Złożone typy danych – łańcuchy. 6) Złożone typy danych – tablice jednowymiarowe. 7) Złożone typy danych – tablice wielowymiarowe. 8) Programowanie obiektowe. 9) Tworzenie własnych klas. 10) Dziedziczenie. 11) Interfejsy i klasy wewnętrzne. 12) Wyjątki i obsługa błędów. 13) Strumienie i pliki. 14) Strumienie obiektów. Równolegle samodzielne przygotowywanie w ramach pracy własnej (przy konsultacyjnym wsparciu prowadzącego) projektu i implementacji prostej aplikacji.

**Metody oceny:**

Wykład: Ocena formatywna: częściowo interaktywna forma prowadzenia wykładu. Ocena sumatywna: przeprowadzenie kolokwium zwierającego pytania otwarte oraz zadania programistyczne; ocena z kolokwium w zakresie 2-5; do zaliczenia wymagane jest uzyskanie oceny >=3.
Laboratorium: Ocena formatywna: na zajęciach weryfikowane jest wykonanie ćwiczeń laboratoryjnych; elementy projektu są dyskuto-wane i weryfikowane na każdym z etapów jego realizacji – jest możliwość poprawienia wyników każdego etapu. Ocena sumatywna: przeprowadzenie kolokwium w formie praktycznej realizacji zadań programistycznych; oceniana jest wartość merytoryczna projektów i uzyskane na ich podstawie wyniki prac implementacyjnych oraz wynik rozmowy zaliczeniowej z prowadzącym; ocena z laboratorium w zakresie 2-5; do zaliczenia wymagane jest uzyskanie oceny >=3.

**Egzamin:**

tak

**Literatura:**

[1] Horstmann C.S., Cornell G.: Java. Podstawy. Wydawnictwo He-lion, Gliwice 2008 . [2] Eckel B.: Thinking in Java. Wydawnictwo Helion, Gliwice 2006. [3] Schildt H.: Java. Kompendium programisty, Wydawnictwo Helion, Gliwice 2005.

**Witryna www przedmiotu:**

http://www.electurer.edu.pl

**Uwagi:**

-

## Efekty przedmiotowe

### Profil ogólnoakademicki - wiedza

**Efekt PODPR\_W01:**

 ma usystematyzowaną wiedzę z zakresu podstaw programowania w języku wysokiego poziomu z położeniem nacisku na programowanie zorientowane obiektowo

Weryfikacja:

1 kolokwium zwierające pytania otwarte oraz zadania (wykład) i 1 kolokwium w formie praktycznej realizacji zadań programistycznych (laboratorium)

**Powiązane efekty kierunkowe:** K\_W12

**Powiązane efekty obszarowe:** S1A\_W06

### Profil ogólnoakademicki - umiejętności

**Efekt PODPR\_U01:**

 potrafi implementować proste algorytmy oraz typowe struktury danych

Weryfikacja:

weryfikacja wykonania ćwiczeń laboratoryjnych, ocena samodzielnie przygotowanego projektu i implementacji aplikacji

**Powiązane efekty kierunkowe:** k\_U10

**Powiązane efekty obszarowe:** S1A\_U07

**Efekt PODPR\_U02:**

 potrafi pisać oraz uruchamiać proste programy z wykorzystaniem zintegrowanego środowiska programistycznego

Weryfikacja:

weryfikacja wykonania ćwiczeń laboratoryjnych, ocena samodzielnie przygotowanego projektu i implementacji aplikacji

**Powiązane efekty kierunkowe:** k\_U10

**Powiązane efekty obszarowe:** S1A\_U07

### Profil ogólnoakademicki - kompetencje społeczne

**Efekt PODPR\_K01:**

 rozumie potrzebę zachowań personalnych i przestrzega zasad etyki, w tym uczciwości

Weryfikacja:

ocena stopnia samodzielności pracy podczas kolokwiów, ocena merytorycznej znajomości przygotowanej aplikacji podczas rozmowy zaliczenio-wej

**Powiązane efekty kierunkowe:** K\_K05

**Powiązane efekty obszarowe:** S1A\_K04