**Nazwa przedmiotu:**

Informatyka I

**Koordynator przedmiotu:**

dr inż. Andrzej Czerepicki, Wydział Transportu Politechniki Warszawskiej, Zakład Systemów Informatycznych i Mechatronicznych w Transporcie

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Transport

**Grupa przedmiotów:**

Obowiązkowe

**Kod przedmiotu:**

TR.SIK203

**Semestr nominalny:**

2 / rok ak. 2017/2018

**Liczba punktów ECTS:**

4

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

114 godz., w tym: praca na wykładach 30 godz., praca na ćwiczeniach laboratoryjnych 30 godz., studiowanie literatury w zakresie wykładu 15 godz., przygotowanie się do egzaminu 5 godz., konsultacje w zakresie wykładu 1 godz., udział w egzaminie 1 godz., przygotowanie się do kolokwiów w zakresie ćwiczeń laboratoryjnych 30 godz., konsultacje w zakresie ćwiczeń laboratoryjnych 3 godz.

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

2,5 pkt. ETCS (65 godz., w tym: praca na wykładach 30 godz., praca na ćwiczeniach laboratoryjnych 30 godz., konsultacje 4 godz. (w tym konsultacje w zakresie ćwiczeń laboratoryjnych 3 godz.), udział w egzaminie 1 godz.)

**Język prowadzenia zajęć:**

polski

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

2,5 pkt. ECTS (63 godz., w tym: praca na ćwiczeniach laboratoryjnych 30 godz., przygotowanie się do kolokwiów w zakresie ćwiczeń laboratoryjnych 30 godz., konsultacje w zakresie ćwiczeń laboratoryjnych 3 godz.)

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

|  |  |
| --- | --- |
| Wykład:  | 30h |
| Ćwiczenia:  | 0h |
| Laboratorium:  | 30h |
| Projekt:  | 0h |
| Lekcje komputerowe:  | 0h |

**Wymagania wstępne:**

brak

**Limit liczby studentów:**

wykład: brak, laboratoria: 16 osób

**Cel przedmiotu:**

Opanowanie zasad programowania w językach wysokiego poziomu, programowanie strukturalne i obiektowe, algorytmy i struktury danych, techniki tworzenia kodów źródłowych: deklarowanie danych, instrukcje i procedury funkcyjne, instrukcje warunkowe, pętle programowe, iteracja; aplikacje konsolowe, aplety oraz aplikacje z interfejsem graficznym.

**Treści kształcenia:**

Wykład:
Wstęp do programowania. Środowisko zintegrowane programowania, zasady pisania kodu źródłowego. Kompilacja i uruchomienie programu. Aplikacja konsolowa. Instrukcje wejścia-wyjścia. Podstawowe typy danych. Operacje arytmetyczne. Pojęcie zmiennej. Instrukcje warunkowe. Instrukcje pętli. Tablice i pliki. Podstawy programowania obiektowego. Klasy, obiekty oraz interfejsy. Metody i pola klasy. Dziedziczenie klas. Aplikacje typu aplet. Podstawy programowania grafiki. Obsługa poleceń klawiatury i myszy. Aplikacje z interfejsem graficznym. Menu, okno główne aplikacji oraz elementy sterowania. Programowanie zdarzeniowe. Równoległe wykonanie kodu. Aplikacje na urządzenia mobilne. Podstawy programowania baz danych. Konfiguracja połączenia z bazą danych, wykonanie prostych poleceń oraz odczyt wyników. Podstawy programowania aplikacji sieciowych.
Ćwiczenia:
Zasady pisania programów w zintegrowanym środowisku NetBeans z wykorzystaniem standardów tworzenia kodów źródłowych i ich przyporządkowania zdarzeniom i zobrazowaniom graficznym. Programowanie aplikacji z interfejsem tekstowym, graficznym oraz apletów WWW. Wykorzystanie różnych własności komponentów środowiska programowego, okien komunikatów i okien dialogowych, operowanie podstawowymi typami zmiennych oraz operatorów i procedur funkcyjnych w zastosowaniach do obliczeń numerycznych i innych aplikacji.

**Metody oceny:**

Wykład: ocena jest wystawiana na podstawie liczby punktów uzyskanych przez studenta na egzaminie; egzamin przeprowadzany jest w formie testu komputerowego składającego się z 15..30 pytań z zakresu zagadnień omawianych na wykładach. W celu zaliczenia egzaminu należy odpowiedzieć poprawnie na min. 51% pytań.
Laboratoria: ocena jest wystawiana na podstawie sumy punktów uzyskanych z dwóch kolokwiów pisemnych. Każde kolokwium jest oceniane w skali od 0 do 50 punktów. Punkty są przyznawane zgodnie z ustaloną listą, która zawiera nazwę ocenianej cechy rozwiązania oraz liczbę przyznawanych punktów. W celu zaliczenia laboratoriów należy uzyskać min. 51 punkt.

**Egzamin:**

tak

**Literatura:**

1) C.S. Horstmann, G. Cornell: Core Java 2 podstawy, Helion, Gliwice, 2003
2) M.Lis: Java ćwiczenia praktyczne, wyd.II, Gliwice 2006
3) K. Rychlicki Kiciom: J2ME Java dla urządzeń mobilnych, Helion, Gliwice , 2006
4) K. Rychlicki Kiciom: J2ME Praktyczne projekty, Helion, 2006

**Witryna www przedmiotu:**

http://www.simt.wt.pw.edu.pl/

**Uwagi:**

## Efekty przedmiotowe

### Profil ogólnoakademicki - wiedza

**Efekt W01:**

Zdobywa wiedzę o zasadach budowy programów komputerowych, typach danych, ich strukturze oraz sposobach deklarowania

Weryfikacja:

wykład - egz. – pisemny, ćwiczenia - kolokwium

**Powiązane efekty kierunkowe:** Tr1A\_W06

**Powiązane efekty obszarowe:** T1A\_W02, InzA\_W05

**Efekt W02:**

Zna zasady tworzenia kodów źródłowych programów komputerowych i ich kompilacji w odniesieniu do programowania strukturalnego i obiektowego

Weryfikacja:

wykład - egz. – pisemny, ćwiczenia - kolokwium

**Powiązane efekty kierunkowe:** Tr1A\_W06

**Powiązane efekty obszarowe:** T1A\_W02, InzA\_W05

**Efekt W03:**

Zna podstawowe koncepcje programowania obiektowego: hermetyzację, dziedziczność, polimorfizm oraz metody programowania obiektowego z wykorzystaniem języka Java

Weryfikacja:

wykład - egz. – pisemny, ćwiczenia - kolokwium

**Powiązane efekty kierunkowe:** Tr1A\_W06

**Powiązane efekty obszarowe:** T1A\_W02, InzA\_W05

**Efekt W04:**

Zna zasady programowania sieciowego w języku Java w odniesieniu do tworzenia prostych programów

Weryfikacja:

wykład - egz. – pisemny, ćwiczenia - kolokwium

**Powiązane efekty kierunkowe:** Tr1A\_W06

**Powiązane efekty obszarowe:** T1A\_W02, InzA\_W05

**Efekt W05:**

Posiada wiedzę z zakresu programowania prostych aplikacji typu aplet oraz servlet oraz ich osadzenia w dokumentach HTML

Weryfikacja:

wykład - egz. – pisemny, ćwiczenia - kolokwium

**Powiązane efekty kierunkowe:** Tr1A\_W06

**Powiązane efekty obszarowe:** T1A\_W02, InzA\_W05

**Efekt W06:**

Zna i rozumie możliwości, ograniczenia oraz zasady operowania językiem Java w środowisku bezprzewodowym

Weryfikacja:

wykład - egz. – pisemny, ćwiczenia - kolokwium

**Powiązane efekty kierunkowe:** Tr1A\_W06

**Powiązane efekty obszarowe:** T1A\_W02, InzA\_W05

### Profil ogólnoakademicki - umiejętności

**Efekt U01:**

Potrafi pozyskiwać informacje dotyczące programowania sieciowego w języku Java z dostępnej literatury, baz danych i źródeł Internetu oraz stosować je w procesie napisania programu komputerowego

Weryfikacja:

wykład - egz. – pisemny, ćwiczenia - kolokwium

**Powiązane efekty kierunkowe:** Tr1A\_U01

**Powiązane efekty obszarowe:** T1A\_U01

**Efekt U02:**

Ma umiejętność samodzielnego pisania prostych programów w języku Java zorientowanych na wykorzystanie w sieci Internet z użyciem technik programowania obiektowego i strukturalnego

Weryfikacja:

wykład - egz. – pisemny, ćwiczenia - kolokwium

**Powiązane efekty kierunkowe:** Tr1A\_U10

**Powiązane efekty obszarowe:** T1A\_U07, T1A\_U09, InzA\_U02

### Profil ogólnoakademicki - kompetencje społeczne

**Efekt K01:**

Rozumie potrzebę systematycznego podnoszenia wiedzy informatycznej wynikłą z szybkiego rozwoju informatyki jej metod i zastosowań

Weryfikacja:

przedłużona obserwacja

**Powiązane efekty kierunkowe:** Tr1A\_K01

**Powiązane efekty obszarowe:** T1A\_K01

**Efekt K02:**

Potrafi identyfikować potrzeby wspomagania informatycznego poprzez programowalne usługi w środowisku Internetu

Weryfikacja:

przedłużona obserwacja

**Powiązane efekty kierunkowe:** Tr1A\_K05

**Powiązane efekty obszarowe:** T1A\_K06, InzA\_K02

**Efekt K03:**

Potrafi poprzez zdobyte umiejętności samodzielnie stosować programowanie usług w środowisku przyszłej, profesjonalnej pracy

Weryfikacja:

przedłużona obserwacja

**Powiązane efekty kierunkowe:** Tr1A\_K05

**Powiązane efekty obszarowe:** T1A\_K06, InzA\_K02