

Object-oriented programming and advanced programming methods - course description

General information	
Course name	Object-oriented programming and advanced programming methods
Course ID	13.2-WI-GeoTSP-Pr.obiekt.-S18
Faculty	Faculty of Civil Engineering, Architecture and Environmental Engineering
Field of study	Geoinformatics and satellite technology
Education profile	academic
Level of studies	First-cycle studies leading to Engineer's degree
Beginning semester	winter term 2021/2022

Course information	
Semester	2
ECTS credits to win	4
Course type	obligatory
Teaching language	polish
Author of syllabus	dr inż. Tomasz Gratkowski

Classes forms					
The class form	Hours per semester (full-time)	Hours per week (full-time)	Hours per semester (part-time)	Hours per week (part-time)	Form of assignment
Lecture	30	2	18	1,2	Exam
Laboratory	30	2	18	1,2	Credit with grade

Aim of the course

Prezentowanie podstawowych pojęć związanych z programowaniem obiektowym (klasa, obiekt, abstrakcja, hermetyzacja, polimorfizm, dziedziczenie) oraz ich realizacją w wykorzystaniu języka programowania Java. Ukształtowanie potrzeby projektowania programów z zastosowaniem wzorców projektowych. Zapoznanie z nowoczesnymi narzędziami programistycznymi wspierającymi wytwarzanie oprogramowania obiektowego na platformie Java.

Prerequisites

Brak

Scope

1. Wprowadzenie - Omówienie środowiska Java Development Kit oraz zintegrowanych środowisk deweloperskich IDE dla platformy Java. Podstawy kompilacji i uruchamiania programów na platformie Java.
2. Algorytmy - Definicje algorytmu, cechy algorytmu, metody zapisu algorytmów, złożoność obliczeniowa
3. Reprezentacja liczb, zmienne - Typy danych, zmienne proste i referencyjne, literały, operatory, tablice, instrukcje sterujące, zasięg widoczności zmiennych, funkcje, właściwości zmiennych.
4. Myślenie w sposób obiektowy - Klasy i instancje, właściwości klas i metod. Zasady konstrukcji obiektów i poznanie mechanizmu czyszczenia pamięci.
5. Dziedziczenie - Dziedziczenie, polimorfizm i hermetyzacja. Projektowanie złożonych typów obiektowych przy użyciu kompozycji i dziedziczenia.
6. Tajniki abstrakcji - Tworzenie interfejsów programistycznych z wykorzystaniem klas abstrakcyjnych i interfejsów. Rozszerzanie interfejsów. Klasy wewnętrzne i klasy statycznie zagnieżdżone.
7. Kolekcje - Przechowywanie złożonych informacji w aplikacji (Listy, kolejki, zbiory, mapy i algorytmy).
8. Obsługa wyjątków & Korzystanie z innych bibliotek - Zabezpieczenie kodu i przechwytywanie wyjątków. Operacje wyjścia/wejścia - co są strumienie.
9. Tworzenie graficznego interfejsu użytkownika

Teaching methods

Wykład - wykład konwencjonalny z wykorzystaniem wideoprojektora.

Laboratorium - zajęcia praktyczne w laboratorium komputerowym.

Learning outcomes and methods of their verification

Outcome description	Outcome symbols	Methods of verification	The class form
Zna zasady projektowania i programowania obiektowego	K_W04	an exam - oral, descriptive, test and other	Lecture
Ma wiedzę na temat procesu wytwarzania oprogramowania w oparciu o technologię Java	K_W04 K_U06	an ongoing monitoring during classes	Laboratory
Zna definicje podstawowych paradygmatów programowania	K_W04	an exam - oral, descriptive, test and other	Lecture

Outcome description	Outcome symbols	Methods of verification	The class form
Potrafi skompilować i uruchomić samodzielnie napisaną aplikację w języku Java	K_U06	an ongoing monitoring during classes	Laboratory

Assignment conditions

Wykład - egzamin w formie pisemnej i/lub w formie testu

Laboratorium – ocena końcowa stanowi sumę ważoną ocen uzyskanych ze sprawdzianów kontrolnych weryfikujących przygotowanie merytoryczne do ćwiczeń oraz za realizację poszczególnych ćwiczeń laboratoryjnych

Ocena końcowa = 50 % oceny zaliczenia z formy zajęć wykład + 50 % oceny zaliczenia z formy zajęć laboratorium.

Recommended reading

Sierra K., Bates B.: Java. Rusz głową! Wydanie II, Helion, 2010

Horstmann, C.S., Cornell, G.: Java., Podstawy, Wydanie dziewiąte, Gliwice, Helion, 2013

Horstmann, C.S., Cornell, G., Java. Techniki zaawansowane, Wydanie dziewiąte, Gliwice, Helion, 2013

Wróblewski, M.: Algorytmy, struktury danych i techniki programowania. Wydanie V, Helion, 2015

Cormen T. H., Leiserson C. E., Rivest R. L., Stein C., Wprowadzenie do algorytmów, Wydawnictwo Naukowe PWN, 2018

Further reading

Martin, R.C.: Czysty kod. Podręcznik dobrego programisty, Helion, 2010

Eckel, B., Thinking in Java, Wydanie IV, Warszawa, Helion, 2006

Lis, M., Praktyczny kurs Java, Wydanie II, Gliwice, Helion, 2004

Coldwind G., Zrozumieć programowanie, Wydawnictwo Naukowe PWN, 2018

Notes

Modified by dr inż. Gerard Bryś (last modification: 16-04-2021 12:50)

Generated automatically from SylabUZ computer system