

Programowanie obiektowe - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Programowanie obiektowe
Kod przedmiotu	11.3-WE-AiRP-PO
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Automatyka i robotyka / Komputerowe systemy sterowania i diagnostyki
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. inżyniera
Semestr rozpoczęcia	semestr zimowy 2016/2017

Informacje o przedmiocie	
Semestr	2
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	dr hab. inż. Paweł Majdzik, prof. UZ

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	30	2	18	1,2	Zaliczenie na ocenę
Laboratorium	30	2	18	1,2	Zaliczenie na ocenę

Cel przedmiotu

- zapoznanie studentów z podstawowymi pojęciami programowania obiektowego: enkapsulacja, klasa, obiekt, metoda składowa
- zapoznanie studentów z metodami hermetyzacji danych, funkcji i klas ukształtowanie wśród studentów umiejętności projektowania programów obiektowych
- zapoznanie studentów z metodami implementacji programów zorientowanych obiektowo: dziedziczenie, polimorfizm

Wymagania wstępne

Programowanie z elementami algorytmiki

Zakres tematyczny

Wstęp do programowania obiektowego. Pojęcie abstrakcyjnego typu danych. Definicja klas. Enkapsulacja - deklaracja i definicja metod składowych klas. Przekazywanie parametrów do funkcji składowych: przez wartość i przez referencję.

Składowe prywatne i publiczne klasy. Przeciążenie funkcji. Konstruktory: konstruktor domniemany, konstruktor kopiujący.

Lista inicjalizacyjna konstruktora. Konstruktory syntezowane. Destruktry. Przeciążenie operatorów. Funkcje zaprzyjaźnione. Funkcje typu inline. Konwersje zdefiniowane przez użytkownika: funkcja konwertująca, konstruktor konwertujący.

Dziedziczenie. Zasady dziedziczenia. Składowe typu protected. Dziedziczenie wielokrotne i wielobazowe. Problem nazw zmiennych w dziedziczeniu wielobazowym.

Polimorfizm. Funkcje wirtualne. Funkcje czysto wirtualne. Wczesne i późne wiązanie funkcji. Koszty czasowe i pamięciowe związane ze stosowaniem polimorfizmu.

Klasy abstrakcyjne. Definiowanie i przykłady zastosowań klas abstrakcyjnych w programach zorientowanych obiektowo. Destruktry wirtualne. Szablony funkcji. Definicja funkcji szablonowych. Funkcje specjalizowane. Etapy dopasowania do funkcji. Szablony klas. Definicja szablonów klas. Szablony klas, a makrodefinicje. Składniki statyczne w szablonie klas. Dziedziczenie szablonów klas. Przykłady zastosowań szablonów klas. Klasy specjalizowane. Obsługa wyjątków.

Stosowanie wzorców z biblioteki STL – wzorce pojemnikowe, iteratory, pojemniki kojarzące (mapy, zbiory).

Metody kształcenia

wykład: wykład konwencjonalny

laboratorium: ćwiczenia laboratoryjne

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
potrafi zastosować gotowe biblioteki klas w swoich programach	• K_W05	• kolokwium	• Laboratorium

Opis efektu	Symbol e efektów	Metody weryfikacji	Forma zajęć
potrafi zaprojektować i zaimplementować diagram klas oraz zastosować polimorfizm	K_W05	kolokwium	Laboratorium
potrafi zdefiniować i zaimplementować podstawowe elementy składowe klasy: konstruktory, funkcje operatorowe, destruktory	K_W05	egzamin - ustny, opisowy, testowy i inne	- Wykład - Laboratorium
rozumie podstawowe pojęcia związane z programowaniem obiektowym: enkapsulację, hermetyzację	K_W05	egzamin - ustny, opisowy, testowy i inne	- Wykład - Laboratorium
rozumie różnicę pomiędzy programowaniem obiektowym i strukturalnym i potrafi zdefiniować wady i zalety obydwu metodyk	K_W05	egzamin - ustny, opisowy, testowy i inne	- Wykład - Laboratorium

Warunki zaliczenia

Wykład – warunkiem zaliczenia jest uzyskanie pozytywnej oceny ze sprawdzianu w formie pisemnej

Laboratorium – warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich sprawdzianów

Składowe oceny końcowej = wykład: 50% + laboratorium: 50%

Literatura podstawowa

- Eckel B.: Thinking in C++, Hellion, Warszawa, 2002.
- Kerighan B., Ritchie D.: Programowanie w języku C, WNT, Warszawa, 2000.
- Kisilewicz J.: Język. w środowisku Borland C++. Wydanie IV, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2003.
- Stroustrup B.: C++ Język programowania, WNT, Warszawa, 2001.

Literatura uzupełniająca

- Lippman S.B.: Model w C++, WNT, Warszawa, 1996.
- Shalloway A., Trott J.R.: Projektowanie zorientowane obiektowo. Wzorce obiektowe II, Helion, Warszawa, 2005.

Uwagi

Zmodyfikowane przez dr hab. inż. Paweł Majdzik, prof. UZ (ostatnia modyfikacja: 12-09-2016 13:26)

Wygenerowano automatycznie z systemu SylabUZ