

POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2015/2016

Wydział Mechaniczny

Kierunek studiów: Mechanika i Budowa Maszyn

Profil: Ogólnoakademicki

Forma studiów: stacjonarne

Kod kierunku: M

Stopień studiów: II

Specjalności: Mechanika Konstrukcji i Materiałów

1 INFORMACJE O PRZEDMIOCIE

NAZWA PRZEDMIOTU	Programowanie obiektowe
NAZWA PRZEDMIOTU W JĘZYKU ANGIELSKIM	Object oriented programming
KOD PRZEDMIOTU	M941
KATEGORIA PRZEDMIOTU	Przedmioty specjalnościowe
LICZBA PUNKTÓW ECTS	1.00
SEMESTRY	2

2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

SEMESTR	WYKŁAD	ĆWICZENIA	LABORATORIUM	LABORATORIUM KOMPUTERO- WE	PROJEKT	SEMINARIUM
2	15	0	0	15	0	0

3 CELE PRZEDMIOTU

Cel 1 Zapoznanie studentów z techniką programowania obiektowo orientowanego na przykładzie języka C++.

4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

1 Zaliczony przedmiot "Podstawy programowania"

2 Zaliczony przedmiot "Praktyka programowania" sem. I

5 EFEKTY KSZTAŁCENIA

EK1 Wiedza Potrafi wymienić podstawowe pojęcia programowania obiektowego OOP.

EK2 Umiejętności Potrafi zakodować w języku C++ model numeryczny prostego zjawiska mechanicznego.

EK3 Umiejętności Potrafi debugować i testować działanie programu.

EK4 Umiejętności Potrafi projektować proste aplikacje modelujące wybrane zagadnienia mechaniczne w technice programowania obiektowo orientowanego OOP (Object Oriented Programming).

6 TREŚCI PROGRAMOWE

LABORATORIUM KOMPUTEROWE		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
K1	Poznanie środowiska pracy Microsoft Visual Studio.	2
K2	Utworzenie projektu OOP modelującego wybrany układ mechaniczny: wyodrębnienie klas, metod i właściwości, określenie relacji i współpracy pomiędzy obiektami. Weryfikacja poprawności modelu.	4
K3	Praca w środowisku Microsoft Visual Studio: implementacja modelu, debugowanie i weryfikacja kodu. Testowanie działania programu.	9

WYKŁAD		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W1	Przypomnienie podstawowych pojęć programowania obiektowego OOP: klasa, enkapsulacja, dziedziczenie, polimorfizm, funkcja wirtualna.	3
W2	Metody projektowania programów techniką OOP: analiza zachowań obiektów, obserwacja obowiązków systemu.	3
W3	Weryfikacja poprawności projektu: tabela relacji, graf współpracy. Istotne aspekty opisu świata rzeczywistego za pomocą obiektów - zapewnienie zasad: uogólnienia (generality), wtórnego użycia (reusability) i rozszerzalności (extensibility).	3
W4	Modele wybranych zagadnień mechanicznych.	3
W5	Zapis do pliku. Korzystanie z biblioteki klas. Środowisko programistyczne Microsoft Visual Studio.	3

7 NARZĘDZIA DYDAKTYCZNE

N1 Wykłady

N2 Ćwiczenia laboratoryjne

8 OBCIĄŻENIE PRACĄ STUDENTA

FORMA AKTYWNOŚCI	ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI
Godziny kontaktowe z nauczycielem akademickim, w tym:	
Godziny wynikające z planu studiów	30
Konsultacje przedmiotowe	0
Egzaminy i zaliczenia w sesji	0
Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym:	
Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	0
Opracowanie wyników	0
Przygotowanie raportu, projektu, prezentacji, dyskusji	0
SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA	30
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	1.00

9 SPOSOBY OCENY

OCENA FORMUJĄCA

F1 Ćwiczenie praktyczne

OCENA PODSUMOWUJĄCA

P1 Projekt

KRYTERIA OCENY

EFEKT KSZTAŁCENIA 1	
NA OCENĘ 2.0	Brak wiedzy wymaganej do uzyskania oceny E.
NA OCENĘ 3.0	Student potrafi objaśnić pojęcia klasa, enkapsulacja, konstruktor, destruktor.

NA OCENĘ 3.5	Student potrafi jak wyżej oraz dodatkowo potrafi objaśnić pojęcia dziedziczenie, polimorfizm, funkcja wirtualna.
NA OCENĘ 4.0	Student potrafi jak wyżej oraz dodatkowo potrafi opisać metody projektowania programów techniką OOP.
NA OCENĘ 4.5	Student potrafi jak wyżej oraz dodatkowo potrafi opisać metody weryfikacji poprawności projektu.
NA OCENĘ 5.0	Student potrafi jak wyżej oraz dodatkowo potrafi wskazać istotne aspekty opisu świata rzeczywistego za pomocą obiektów.
EFEKT KSZTAŁCENIA 2	
NA OCENĘ 2.0	Brak wiedzy wymaganej do uzyskania oceny E.
NA OCENĘ 3.0	Student potrafi przepisać podany kod w języku C++ .
NA OCENĘ 3.5	Student potrafi jak wyżej oraz dodatkowo tworzy fragmenty kodu z podstawowymi konstrukcjami programistycznymi, jak: pętle, wyrażenia warunkowe, etc.
NA OCENĘ 4.0	Student potrafi jak wyżej oraz dodatkowo tworzy fragmenty kodu w postaci deklaracji i definicji funkcji.
NA OCENĘ 4.5	Student potrafi jak wyżej oraz dodatkowo tworzy fragmenty kodu implementacji klas.
NA OCENĘ 5.0	Student potrafi jak wyżej oraz dodatkowo tworzy obiekty klas i posługuje się ich składowymi oraz metodami.
EFEKT KSZTAŁCENIA 3	
NA OCENĘ 2.0	Brak wiedzy wymaganej do uzyskania oceny E.
NA OCENĘ 3.0	Student potrafi zbudować i uruchomić program.
NA OCENĘ 3.5	Student potrafi jak wyżej oraz dodatkowo posługuje się pewnymi elementami narzędzia debugger.
NA OCENĘ 4.0	Student potrafi jak wyżej oraz dodatkowo posługuje się narzędziem debugger.
NA OCENĘ 4.5	Student potrafi jak wyżej oraz dodatkowo posługuje się pewnymi elementami narzędzia profiler.
NA OCENĘ 5.0	Student potrafi jak wyżej oraz dodatkowo posługuje się narzędziem profiler.
EFEKT KSZTAŁCENIA 4	
NA OCENĘ 2.0	Brak wiedzy wymaganej do uzyskania oceny E.
NA OCENĘ 3.0	Student potrafi opisać relacje w podanym projekcie OOP.
NA OCENĘ 3.5	Student potrafi jak wyżej oraz dodatkowo potrafi przeprowadzić analizę poprawności projektu OOP.

NA OCENĘ 4.0	Student potrafi jak wyżej oraz dodatkowo potrafi wskazać klasy w projekcie modelującym wybrane zagadnienie mechaniczne.
NA OCENĘ 4.5	Student potrafi jak wyżej oraz dodatkowo potrafi wskazać właściwości i metody obiektów.
NA OCENĘ 5.0	Student potrafi jak wyżej oraz dodatkowo wykonuje projekt programu techniką OOP.

10 MACIERZ REALIZACJI PRZEDMIOTU

EFEKT KSZTAŁCENIA	ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓLOWYCH EFEKTÓW ZDEFINIOWANYCH DLA PROGRAMU	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	NARZĘDZIA DYDAKTYCZNE	SPOSOBY OCENY
EK1	K2_W15	Cel 1	W1 W2 W3 W4 W5	N1 N2	P1
EK2	K2_UB10	Cel 1	K1 K2 K3	N2	F1
EK3	K2_UB10	Cel 1	K1 K2 K3	N2	F1
EK4	K2_UB10	Cel 1	K1 K2 K3	N2	F1 P1

11 WYKAZ LITERATURY

LITERATURA PODSTAWOWA

- [1] Grębosz J. — *Symfonia C++*, Kraków, 1996, Oficyna Kallimach
 [2] Stasiewicz A. — *C++ ćwiczenia zaawansowane*, Gliwice, 2005, Helion

LITERATURA UZUPEŁNIAJĄCA

- [1] Wirth N. — *Algorytmy + struktury danych = programy*, , 1983, WNT
 [2] Yang D. — *C++ and object-oriented numeric computing for scientists and engineers*, New York, 2001, Springer-Verlag

12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

OSOBA ODPOWIEDZIALNA ZA KARTĘ

dr inż. Paweł Foryś (kontakt: pforys@pk.edu.pl)



OSOBY PROWADZĄCE PRZEDMIOT

1 dr inż. Władysław Egner (kontakt: wladyslaw.egner@pk.edu.pl)

2 dr inż. Jan Bielski (kontakt: jan.bielski@pk.edu.pl)

13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejsowość, data)

(odpowiedzialny za przedmiot)

(dziekan)

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....

.....