

POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2018/2019

Wydział Mechaniczny

Kierunek studiów: Mechanika i Budowa Maszyn

Profil: Ogólnoakademicki

Forma studiów: niestacjonarne

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	WM MIBM oIIN D11 18/19
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	9	0	0	9	0	0

3 CELE PRZEDMIOTU

Cel 1 Zapoznanie studentów z technikami programowania obiektowo orientowanego z przykładami w języku Java.

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 Student zna i potrafi wyjaśnić podstawowe pojęcia programowania obiektowego.

EK2 Umiejętności Student potrafi napisać prosty program zdefiniowany za pomocą obiektów łączących stan (pola) i zachowanie (metody).

EK3 Umiejętności Student potrafi w sposób systematyczny redukować liczbę błędów w oprogramowaniu poprzez kontrolowanie wykonania programu pod nadzorem debuggera.

EK4 Umiejętności Student potrafi projektować proste aplikacje, modelujące obiektowo wybrane zagadnienia oraz zależności między nimi.

6 TREŚCI PROGRAMOWE

WYKŁAD		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W1	Omówienie podstawowych paradygmatów programowania obiektowego: klasa, obiekt, dziedziczenie, polimorfizm.	3
W2	Metody projektowania programów obiektowo zorientowanych: diagram klas, podstawy UML.	3
W3	Korzystanie z bibliotek klas. Zintegrowane środowisko programistyczne (IDE) dla języka Javy: IntelliJ IDEA.	3

LABORATORIUM KOMPUTEROWE		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
K1	Poznanie środowiska pracy (IntelliJ IDEA).	2
K2	Utworzenie projektu modelującego wybrane zagadnienie rzeczywiste: wyodrębnienie klas, metod i właściwości, określenie relacji i współpracy pomiędzy obiektami. Weryfikacja poprawności modelu.	3
K3	Praca w środowisku IntelliJ: implementacja modelu, debugowanie i weryfikacja kodu. Testowanie działania programu.	4

7 NARZĘDZIA DYDAKTYCZNE

N1 Wykłady

N2 Laboratoria komputerowe

8 OBCIĄŻENIE PRACĄ STUDENTA

FORMA AKTYWNOŚCI	ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI
Godziny kontaktowe z nauczycielem akademickim, w tym:	
Godziny wynikające z planu studiów	18
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	4
Opracowanie wyników	0
Przygotowanie raportu, projektu, prezentacji, dyskusji	8
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 wyjaśnić pojęcia klasa, obiekt, konstruktor, destruktor.
NA OCENĘ 3.5	Student potrafi jak wyżej oraz dodatkowo potrafi wyjaśnić pojęcia dziedziczenie i polimorfizm.

NA OCENĘ 4.0	Student potrafi jak wyżej oraz dodatkowo potrafi opisać metody projektowania programów orientowanych obiektowo.
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 i zinterpretować przykładowy kod w języku Java.
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 utworzyć 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 przykładowym projekcie obiektowym.
NA OCENĘ 3.5	Student potrafi jak wyżej oraz dodatkowo potrafi przeprowadzić analizę poprawności projektu obiektowego.
NA OCENĘ 4.0	Student potrafi jak wyżej oraz dodatkowo potrafi wskazać klasy w projekcie modelującym wybrane zagadnienie rzeczywiste.

NA OCENĘ 4.5	Student potrafi jak wyżej oraz dodatkowo potrafi wskazać własności i metody obiektów.
NA OCENĘ 5.0	Student potrafi jak wyżej oraz tworzy projekty techniką programowania obiektowego.

10 MACIERZ REALIZACJI PRZEDMIOTU

EFEKT KSZTAŁCENIA	ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁOWYCH EFEKTÓW ZDEFINIOWANYCH DLA PROGRAMU	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	NARZĘDZIA DYDAKTYCZNE	SPOSOBY OCENY
EK1	K2_W07 K2_W11 K2_W15	Cel 1	W1 W2 W3	N1 N2	F1 P1
EK2	K2_UB10 K2_UP08	Cel 1	K1 K2 K3	N2	F1
EK3	K2_UB10 K2_UP07	Cel 1	K1 K2 K3	N2	F1
EK4	K2_UB10 K2_UP05 K2_UP08 K2_UP12	Cel 1	K1 K2 K3	N2	F1 P1

11 WYKAZ LITERATURY

LITERATURA PODSTAWOWA

[1] | Bruce Eckelo — *Thinking in Java*, Gliwice, 2006, Helion

[2] | Cay S. Horstmann, Gary Cornell — *Java Podstawy*, Gliwice, 2013, Helion

LITERATURA UZUPEŁNIAJĄCA

[1] | Cay S. Horstmann, Gary Cornell — *Java Techniki zaawansowane*, Gliwice, 2013, Helion

LITERATURA DODATKOWA

[1] | Dan Pilone, Neil Pitman — *UML 2.0. Almanach*, Gliwice, 2012, Helion

12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

OSOBA ODPOWIEDZIALNA ZA KARTĘ

dr inż. Daniel, Tomasz Ziemiański (kontakt: daniel.ziemianski@pk.edu.pl)

OSOBY PROWADZĄCE PRZEDMIOT

1 dr inż. Daniel Ziemiański (kontakt: dziemianski@pk.edu.pl)

13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejscowość, data)

(odpowiedzialny za przedmiot)

(dziekan)

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....