

POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2020/2021

Wydział Mechaniczny

Kierunek studiów: Informatyka Stosowana

Profil: Ogólnoakademicki

Forma studiów: stacjonarne

Kod kierunku: S

Stopień studiów: I

Specjalności: Bez specjalności

1 INFORMACJE O PRZEDMIOCIE

NAZWA PRZEDMIOTU	Programowanie obiektowe
NAZWA PRZEDMIOTU W JĘZYKU ANGIELSKIM	Object-oriented programming
KOD PRZEDMIOTU	WM INFST oIS B7 20/21
KATEGORIA PRZEDMIOTU	Przedmioty kierunkowe
LICZBA PUNKTÓW ECTS	3.00
SEMESTRY	4

2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

SEMESTR	WYKŁAD	ĆWICZENIA	LABORATORIUM	LABORATORIUM KOMPUTERO- WE	PROJEKT	SEMINARIUM
4	15	0	0	30	0	0

3 CELE PRZEDMIOTU

Cel 1 Przekazanie wiedzy i umiejętności z zakresu programowania obiektowego

4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

5 EFEKTY KSZTAŁCENIA

EK1 Wiedza Student zna i rozumie paradygmat programowania obiektowego.

EK2 Wiedza Student zna i rozumie składnię i semantykę języka JAVA.

EK3 Wiedza Student zna i rozumie strukturę i usługi bibliotek wspierających programowanie obiektowe.

EK4 Umiejętności Student potrafi wykorzystać zdobytą wiedzę do programowania aplikacji w modelu obiektowym.

6 TREŚCI PROGRAMOWE

WYKŁAD		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN

WYKŁAD		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W1	Wprowadzenie do obiektowej analizy problemu informatycznego. Obiektowy paradygmat programowania, w tym założenia SOLID. Podstawowe pojęcia i terminy: abstrakcja, enkapsulacja, dziedziczenie, polimorfizm. Inne techniki obiektowe: przeciążanie operatorów, klasy generyczne, RTTI. Przedstawienie środowiska programowania oraz wymaganych narzędzi i bibliotek. Metodyka definiowania klas, atrybutów i metod. Włączanie bibliotek, używanie przestrzeni nazw. Tworzenie obiektów. Składniki klas o specjalnym znaczeniu: konstruktory i destruktory, składniki statyczne, składniki stałe, referencje do obiektów innych klas, klasy zagnieżdżone. Praktyka stosowania enkapsulacji, definiowanie widoczności składników klasy. Obiektowe struktury danych, w tym klasy kontenerowe (kolekcje). Technika dziedziczenia. Charakterystyka i podstawowe rodzaje, w tym dziedziczenie wielobazowe i wielopokoleniowe. Korzyści oraz ograniczenia praktycznego zastosowania tej techniki. Dziedziczenie w kontekście widoczności składników klasy, omówienie składników, które nie są dziedziczone. Hierarchia dziedziczenia klas w bibliotekach SDK. Definiowanie klas i metod wirtualnych. Polimorfizm metody jako technika wywoływania funkcji wirtualnych. Definicja klas abstrakcyjnych oraz tworzenie hierarchii klas z ich udziałem. Mechanizm RTTI jako źródło informacji o klasie w warunkach stosowania polimorfizmu. Projektowanie graficznego interfejsu użytkownika GUI z wykorzystaniem obiektowych bibliotek komponentów wizualnych. Obiektowe techniki obsługi zdarzeń z wykorzystaniem konstrukcji jak wyrażenia lambda. Definicje oraz klasyfikacja sytuacji wyjątkowych. Wykrycie oraz reakcja na wystąpienie wyjątku z wykorzystaniem dostępnych klas. Praktyczna realizacja wymagań obsługi lub określ. Tworzenie klas wyjątków, programowe zgłaszanie wyjątków. Interfejs jako element programowania obiektowego. Definiowanie składników interfejsu. Technika implementacji w klasie oraz użycie w postaci typu referencyjnego jako przykłady praktycznych zastosowań interfejsów. Interfejsy jako alternatywa dla dziedziczenia. Zaawansowane techniki zastosowań interfejsów: interfejsy funkcjonalne, domyślne implementacje, wstrzykiwanie zależności, odwrócone sterowanie. Technika tworzenia klas generycznych z wykorzystaniem uogólnionych typów danych jako przykład polimorfizmu klasy. Praktyka tworzenia klas i metod generycznych z uwzględnieniem korzyści oraz ograniczeń. Zawężenie zakresu typów uogólnionych w klasach generycznych, w tym zastosowanie typu nieokreślonego z ograniczeniem od góry lub od dołu. Przeciążanie operatorów. Wielowątkowość, tworzenie i wykorzystywanie klas implementujących wątki drugoplanowe. Wstęp do wzorców projektowych.	15

LABORATORIUM KOMPUTEROWE		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN

LABORATORIUM KOMPUTEROWE		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
K1	Zapoznanie się ze środowiskiem programowania: edytor kodu, debugger, system pomocy, struktura projektu, pliki źródłowe, pośrednie i wynikowe. Opanowanie podstaw budowy, kompilacji i uruchamiania aplikacji w trybie tekstowym. Zapoznanie się z systemem tworzenia dokumentacji projektowej i jego praktyczne wykorzystanie. Budowa aplikacji wymagającej zdefiniowania klasy i utworzenia kilku instancji. Określanie stanu obiektu podczas jego tworzenia (konstruktor) oraz za pomocą zdefiniowanych metod publicznych. Formatowanie danych wyjściowych. Budowa aplikacji z wykorzystaniem związków klas typu agregacja (zawieranie) oraz asocjacja (zastosowanie kolekcji). Budowa aplikacji z zastosowaniem związku klas typu uogólnienie (technika dziedziczenia). Zapoznanie się z podstawami budowy aplikacji z graficznym interfejsem użytkownika. Wybór układu (layout), rozmieszczanie komponentów, modyfikowanie ich właściwości i obsługa zdarzeń. Budowa aplikacji z GUI oraz praktycznym wykorzystaniem techniki definiowania klas abstrakcyjnych i polimorfizmu metody. Budowa aplikacji wykorzystującej system obsługi wyjątków. Implementacja odpowiedniej, adekwatnej reakcji na wystąpienie sytuacji wyjątkowej oraz programowe zgłaszanie wyjątku. Budowa aplikacji w grupie projektowej z wykorzystaniem techniki definiowania i implementacji interfejsów. Budowa aplikacji w grupie projektowej z wykorzystaniem technik obiektowych, jak wstrzykiwanie zależności i odwrócone sterowanie. Zapoznanie się z obiektowymi narzędziami do współpracy z systemem plików. Budowa aplikacji z wykorzystaniem obiektowych technik zapisu i odczytu danych z plików dyskowych oraz obsługi sytuacji wyjątkowych. Budowa aplikacji implementującej wybrane podstawowe wzorce projektowe z grupy wzorców konstrukcyjnych. Budowa aplikacji realizującej komunikację sieciową z wykorzystaniem gniazd (sockets). Opracowanie protokołu przesyłania danych, implementacja strony serwera i klienta. Budowa aplikacji implementującej wybraną metodę numeryczną z wykorzystaniem technik tworzenia wątków drugoplanowych, przesyłania danych, komunikacji między wątkami i synchronizacji. Budowa aplikacji wykorzystującej obiektowy interfejs na przykładzie komunikacji z serwerem bazodanowym. Uzyskanie połączenia ze zdalną bazą, odczyt i prezentacja oraz zapis danych do bazy.	30

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	45
Konsultacje przedmiotowe	4
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	11
Opracowanie wyników	0
Przygotowanie raportu, projektu, prezentacji, dyskusji	30
SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA	90
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	3.00

9 SPOSOBY OCENY

OCENA FORMUJĄCA

F1 Test z wykładu

F2 Ćwiczenie praktyczne

OCENA PODSUMOWUJĄCA

P1 Średnia ważona ocen formujących

WARUNKI ZALICZENIA PRZEDMIOTU

W1 Pozytywna ocena z wykładu

W2 Pozytywne oceny z laboratoriów

W3 Obecność studenta na min. 75% zajęć laboratoryjnych

KRYTERIA OCENY

EFEKT KSZTAŁCENIA 1	
NA OCENĘ 3.0	Student zna i rozumie w podstawowym zakresie paradygmat programowania obiektowego.
EFEKT KSZTAŁCENIA 2	

NA OCENĘ 3.0	Student zna i rozumie w podstawowym zakresie składnię i semantykę języka JAVA.
EFEKT KSZTAŁCENIA 3	
NA OCENĘ 3.0	Student zna i rozumie w podstawowym zakresie strukturę i usługi bibliotek wspierających programowanie obiektowe.
EFEKT KSZTAŁCENIA 4	
NA OCENĘ 3.0	Student potrafi wykorzystać zdobytą wiedzę do programowania prostej aplikacji w modelu obiektowym.

10 MACIERZ REALIZACJI PRZEDMIOTU

EFEKT KSZTAŁCENIA	ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁOWYCH EFEKTÓW ZDEFINIOWANYCH DLA PROGRAMU	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	NARZĘDZIA DYDAKTYCZNE	SPOSOBY OCENY
EK1		Cel 1	W1 K1	N1 N2	F1 F2 P1
EK2		Cel 1	W1 K1	N1 N2	F1 F2 P1
EK3		Cel 1	W1 K1	N1 N2	F1 F2 P1
EK4		Cel 1	W1 K1	N1 N2	F1 F2 P1

11 WYKAZ LITERATURY

12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

OSOBA ODPOWIEDZIALNA ZA KARTĘ

dr hab. inż., prof. PK Jacek Pietraszek (kontakt: jacek.pietraszek@pk.edu.pl)

OSOBY PROWADZĄCE PRZEDMIOT

1 pracownicy Instytutu Informatyki Stosowanej (kontakt:)

13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejscowość, data)

(odpowiedzialny za przedmiot)

(dziekan)



PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....