

POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2013/2014

Wydział Fizyki, Matematyki i Informatyki

Kierunek studiów: Informatyka

Profil: Ogólnoakademicki

Forma studiów: stacjonarne

Kod kierunku: I

Stopień studiów: I

Specjalności: Brak specjalności

1 INFORMACJE O PRZEDMIOCIE

NAZWA PRZEDMIOTU	Programowanie w języku Java
NAZWA PRZEDMIOTU W JĘZYKU ANGIELSKIM	
KOD PRZEDMIOTU	WFMiI I oIS D1 13/14
KATEGORIA PRZEDMIOTU	Przedmioty specjalnościowe
LICZBA PUNKTÓW ECTS	4.00
SEMESTRY	4

2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

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

3 CELE PRZEDMIOTU

Cel 1 Poznanie składni, zasad obiektowości, obsługi wyjątków. Nabycie umiejętności wykorzystywania IDE (Netbeans). Nabycie umiejętności obiektowego zapisywania prostych algorytmów.

Cel 2 Zapoznanie się z programowaniem wielowątkowym. Nabycie umiejętności programowania wielowątkowego wybranych zagadnień z metod numerycznych.

Cel 3 Zapoznanie się z programowaniem sterowanym zdarzeniami. Nabycie umiejętności tworzenia GUI z wykorzystaniem pakietu Swing.

Cel 4 Poznanie programowania sterowanego czasem. Nabycie umiejętności tworzenia prostych animacji.

4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

1 Umiejętność budowania algorytmów. Znajomość jednego języka programowania (strukturalnego lub obiektowego). Zaliczenie przedmiotów: "Algorytmy i struktury danych" oraz "Języki i paradygmaty programowania".

5 EFEKTY KSZTAŁCENIA

EK1 Wiedza Instrukcje i struktury danych w Javie. Zasady obiektowości. Pojęcia: klasa, obiekt, dziedziczenie, polimorfizm, klasa abstrakcyjna, interfejs, klasa wewnętrzna, wyjątki.

EK2 Umiejętności Budowanie programów wykorzystujących dziedziczenie i polimorfizm. Praktyczne wykorzystanie klas abstrakcyjnych i interfejsów. Wykorzystanie klas wewnętrznych. Zastosowanie klas wyjątków, tworzenie własnych klas wyjątków. Przechwytywanie wyjątków w przykładach.

EK3 Wiedza Klasy wewnętrzne. Anonimowe klasy wewnętrzne. Parametryzacja klas, metod i konstruktorów. Programowanie wielowątkowe. Programowanie sterowane zdarzeniami i czasem. Kolekcje.

EK4 Umiejętności Praktyczne wykorzystanie wielowątkowości w zadaniach numerycznych i w programowaniu sterowanym zdarzeniami oraz tworzeniu prostych animacji.

6 TREŚCI PROGRAMOWE

WYKŁAD		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W1	Przegląd niezbędnego oprogramowania, jego dostępność, instalacja i konfiguracja w systemach Windows i Linux. Dostępne IDE: Netbeans, Eclipse. Podstawowe pojęcia obiektowego programowania w Javie. Historia i cechy języka Java. Maszyna wirtualna Javy. Pojęcie bytecode'u. Przegląd pakietów Javy. Aplikacja, servlet, applet, midlet.	2
W2	Pierwszy program i applet w Javie, kompilacja, wykonanie. Dokumentowanie programów. Klasa, obiekt, składowe klasy, konstruktory, metody. Dostępność składowych i metod. Modyfikatory dostępu, statyczność składowych i metod. Referencje. Korzystanie z dokumentacji. Elementy języka: literały, zmienne proste, obiekty, operatory, instrukcje sterujące.	2
W3	Tablice, przegląd pakietu Arrays, realizacja wejścia i wyjścia, pętla for-each izakres jej wykorzystania w Javie. Wyjątki czasu wykonania. Wykorzystanie parametryzacji. Tablice wieloindeksowe.	2
W4	Dziedziczenie, kolejność wykonania konstruktorów w hierarchii klas. Konstruktory bezargumentowe i z argumentami w hierachii klas. Słowa kluczowe super i this w odniesieniu do konstruktorów, metod i składowych klas. Klasy i metody abstrakcyjne, interfejsy i przykłady ich wykorzystania. Pojęcie sprzątnięcia w Javie wraz z jego ilustracją w przykładzie. Rola metody finalize.	2

WYKŁAD		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W5	Polimorfizm, rzutowanie w górę hierarchii klas, możliwości rzutowania w dół hierarchii klas. Implementacja interfejsów i dziedziczenie z klas abstrakcyjnych. Wykonanie metod na rzecz obiektów o referencjach klasy interfejsu i klasy abstrakcyjnej.	2
W6	Obsługa wyjątków w Javie, przechwytywanie wyjątków, tworzenie klas wyjątków, wyrzucanie wyjątków, hierarchia klas wyjątków. Blok prób sekcja catch i finally. Wyjątki klasy RunTime.	2
W7	Klasy wewnętrzne, anonimowe klasy wewnętrzne. Klasy wewnętrzne a rzutowanie w górę. Klasy wewnętrzne, a ukrywanie implementacji. Klasy wewnętrzne a przejmowanie implementacji z wielu klas. Klasy wewnętrzne i anonimowe klasy wewnętrzne po kompilacji.	2
W8	Wątki, tworzenie klas wątków. Metody klasy wątku (Thread i interfejsu Runnable). Wątki jako klasy wewnętrzne. Wykorzystanie wątków w programowaniu opartym na zdarzeniach. Priorytety wątków. Synchronizacja wątków i obiektów. Komunikacja międzywątkowa.	2
W9	Automatyczne otaczanie i wydobywanie typów prostych, klasy sparametryzowane z dowolną liczbą parametrów ich rola w uproszczeniu programowania, ograniczanie typów w parametrach, argumenty wieloznaczne i ograniczone argumenty wieloznaczne, metody i konstruktory sparametryzowane, interfejsy sparametryzowane. Wyliczenia w Javie, wyliczenia jako typ klasy, klasa Enum jako bazowa klasa dla wyliczeń.	2
W10	Kolekcje: interfejs Collection, interfejs List, klasy ArrayList i LinkedList, metody i przykłady wykorzystania.	2
W11	Interfejs Set, interfejs SortedSet, klasy: HashSet, LinkedHashSet, TreeSet, metody i przykłady wykorzystania.	2
W12	Interfejs Map, klasy: HashMap, TreeMap, metody i przykłady wykorzystania.	2
W13	Aplety, szkielet apletu, przykład prostej animacji z wykorzystywaniem wątków, klasy zdarzeń, źródła zdarzeń, interfejsy nasłuchujące, model zdarzeń. Adaptery. Okna klasy Frame tworzone z poziomu apletu.	2
W14	Pakiet AWT (Abstract Toolkit Window). Aplikacja interaktywna zbudowana za pomocą pakietu AWT. Kontrolki GUI z pakietu AWT, menadżery układu, zastosowanie w apletach i aplikacjach. Kontrolki z pakietu Swing i ich wykorzystanie w apletach i aplikacjach, przykładowe menadżery układu.	2
W15	Kontenery GUI i ich zastosowanie.	2

LABORATORIUM		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN

LABORATORIUM		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
L1	Wykonanie programu z zastosowaniem podstawowych elementów języka (klasa, konstruktor, metoda, składowe klasy, dostępność metod i składowych).	2
L2	Wykonanie programu wykorzystującego dziedziczenie (klasa abstrakcyjna, interfejs, implementacja interfejsu, dziedziczenie, rzutowanie, konstruktory w hierarchii klas, polimorfizm)	4
L3	Wykonanie aplikacji, apletu, midletu tworzących prostą grafikę.	4
L4	Wykonanie programu sterowanego zdarzeniami.	2
L5	Opracowanie programu wielowątkowego.	2
L6	Opracowanie programu z wykorzystaniem komponentów biblioteki Swing.	2
L7	Opracowanie programu z prostą animacją.	4
L8	Opracowanie programu z prostą animacją i sterowaniem zdarzeniami.	4
L9	Opracowanie programów (2-3) z wykorzystaniem komponentów biblioteki Swing (programy typu kalkulator, przeglądarka plików graficznych wraz z prostymi przekształceniami obrazów itp.)	6

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	0
Konsultacje przedmiotowe	20
Egzaminy i zaliczenia w sesji	10
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	20
SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA	50
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	4.00

9 SPOSOBY OCENY

OCENA FORMUJĄCA

F1 Ćwiczenie praktyczne

F2 Kolokwium

F3 Odpowiedź ustna

OCENA PODSUMOWUJĄCA

P1 Średnia ważona ocen formujących

KRYTERIA OCENY

EFEKT KSZTAŁCENIA 1	
NA OCENĘ 2.0	Niezajomość składni języka Java.
NA OCENĘ 3.0	Znajomość składni języka java. Znajomość następujących pojęć: klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu. Hierachia klas. Rzutowanie. Polimorfizm. (80 - 90% poprawnych odpowiedzi)
NA OCENĘ 3.5	Znajomość składni języka java. Znajomość następujących pojęć: klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu. Hierachia klas. Rzutowanie. Polimorfizm. (90 - 100% poprawnych odpowiedzi).

NA OCENĘ 4.0	Znajomość składni języka java. Znajomość następujących pojęć: klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu. Hierachia klas. Rzutowanie. Polimorfizm (90 - 100% poprawnych odpowiedzi). Wyjątki, przechwytywanie wyjątków (60-70% poprawnych odpowiedzi).
NA OCENĘ 4.5	Znajomość składni języka java. Znajomość następujących pojęć: klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu. Hierachia klas. Rzutowanie. Polimorfizm (90 - 100% poprawnych odpowiedzi). Wyjątki, przechwytywanie wyjątków (70-90% poprawnych odpowiedzi).
NA OCENĘ 5.0	Znajomość składni języka java. Znajomość następujących pojęć: klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu. Hierachia klas. Rzutowanie. Polimorfizm (90 - 100% poprawnych odpowiedzi). Wyjątki, przechwytywanie wyjątków (90-100% poprawnych odpowiedzi).
EFEKT KSZTAŁCENIA 2	
NA OCENĘ 2.0	Brak umiejętności programowania zadań o niewielkiej złożoności.
NA OCENĘ 3.0	Praktyczne wykorzystanie pojęć (2-3 programów): klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu, hierachia klas, rzutowanie, polimorfizm. Oddanie programów z dwutygodniowym opóźnieniem z istotnymi usterkami w programie lub niepełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
NA OCENĘ 3.5	Praktyczne wykorzystanie pojęć (2-3 programów): klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu, hierachia klas, rzutowanie, polimorfizm. Oddanie programów z dwutygodniowym opóźnieniem i pełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
NA OCENĘ 4.0	Praktyczne wykorzystanie pojęć (2-3 programów): klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu, hierachia klas, rzutowanie, polimorfizm. Oddanie programów z tygodniowym opóźnieniem z istotnymi usterkami w programie lub niepełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
NA OCENĘ 4.5	Praktyczne wykorzystanie pojęć (2-3 programów): klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu, hierachia klas, rzutowanie, polimorfizm. Oddanie programów w terminie bez istotnych usterek w programie ale niepełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
NA OCENĘ 5.0	Praktyczne wykorzystanie pojęć (2-3 programów): klasa, obiekt, klasa abstrakcyjna, dziedziczenie, interfejs, implementacja interfejsu, hierachia klas, rzutowanie, polimorfizm. Oddanie programów w terminie bez istotnych usterek w programie i pełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
EFEKT KSZTAŁCENIA 3	
NA OCENĘ 2.0	Znajomość zaawansowanych pojęć Javy na poziomie niższym niż 50%.

NA OCENĘ 3.0	Znajomość pojęć: klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje i ich wykorzystanie (znajomość na poziomie 50-60%).
NA OCENĘ 3.5	Znajomość pojęć: klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje i ich wykorzystanie (znajomość na poziomie 60-70%).
NA OCENĘ 4.0	Znajomość pojęć: klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje i ich wykorzystanie (znajomość na poziomie 70-80%).
NA OCENĘ 4.5	Znajomość pojęć: klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje i ich wykorzystanie (znajomość na poziomie 80-90%).
NA OCENĘ 5.0	Znajomość pojęć: klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje i ich wykorzystanie (znajomość na poziomie 90-100%)
EFEKT KSZTAŁCENIA 4	
NA OCENĘ 2.0	Brak umiejętności tworzenia prostych aplikacji.
NA OCENĘ 3.0	Praktyczne wykorzystanie pojęć (5-8 programów): klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje. Oddanie programów z dwutygodniowym opóźnieniem z istotnymi usterkami w programie lub niepełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
NA OCENĘ 3.5	Praktyczne wykorzystanie pojęć (5-8 programów): klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje. Oddanie programów z dwutygodniowym opóźnieniem i pełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
NA OCENĘ 4.0	Praktyczne wykorzystanie pojęć (5-8 programów): klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje. Oddanie programów z tygodniowym opóźnieniem z istotnymi usterkami w programie lub niepełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
NA OCENĘ 4.5	Praktyczne wykorzystanie pojęć (5-8 programów): klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje. Oddanie programów w terminie bez istotnych usterek w programie ale niepełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.

NA OCENĘ 5.0	Praktyczne wykorzystanie pojęć (5-8 programów): klasa wewnętrzna, anonimowa klasa wewnętrzne, parametryzacja klas, metod i konstruktorów, programowanie wielowątkowe, programowanie sterowane zdarzeniami i czasem, kolekcje. Oddanie programów w terminie bez istotnych usterek w programie i pełnymi odpowiedziami na pytania zadane podczas sprawdzania zadań.
--------------	---

10 MACIERZ REALIZACJI PRZEDMIOTU

EFEKT KSZTAŁCENIA	ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁOWYCH EFEKTÓW ZDEFINIOWANYCH DLA PROGRAMU	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	NARZĘDZIA DYDAKTYCZNE	SPOSOBY OCENY
EK1	I1_W06, I1_W07, I1_U02, I1_U05, I1_U07	Cel 1	W1 W2 W3 W4 W5 W6 L1 L2 L3	N1 N2	F2 F3
EK2	I1_W06, I1_W07, I1_U02, I1_U05, I1_U07	Cel 2	W1 W2 W3 W4 W5 W6 L1 L2 L3	N1	F1 F3
EK3	I1_W06, I1_W07, I1_U02, I1_U05, I1_U07	Cel 3	W7 W8 W9 W10 W11 W12 W13 W14 W15 L4 L5 L6 L7 L8 L9	N1 N2	F2 F3
EK4	I1_W06, I1_W07, I1_U02, I1_U05, I1_U07	Cel 4	W7 W8 W9 W10 W11 W12 W13 W14 W15 L4 L5 L6 L7 L8 L9	N1 N2	F1 F3

11 WYKAZ LITERATURY

LITERATURA PODSTAWOWA

- [1] Cay S. Horstmam — *Java Podstawy*, Helion, 2008, Helion
 [2] H. Schild H. Schild — *Java Kompendium programisty*, Gliwice, 2005, Helion

LITERATURA UZUPEŁNIAJĄCA

- [1] Cay S. Horstmam — *Java Techniki zaawansowane*, Warszawa, 2009, Helion

12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

OSOBA ODPOWIEDZIALNA ZA KARTĘ

dr inż. Danuta Zboś (kontakt: zbos@pk.edu.pl)

OSOBY PROWADZĄCE PRZEDMIOT

1 dr Danuta Zboś (kontakt: zbos@pk.edu.pl)

13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejsowość, data)

(odpowiedzialny za przedmiot)

(dziekan)

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....