

## I. KARTA PRZEDMIOTU

Nazwa przedmiotu/modułu:		Wprowadzenie do programowania obiektowego			Kod przedmiotu WPOB	
Nazwa angielska:		Introduction to object oriented programming				
Kierunek studiów:		Edukacja techniczno-informatyczna				
Tryb/Poziom studiów:		Stacjonarne /I-go stopnia – inżynierskie				
Profil studiów		praktyczny				
Jednostka prowadząca:		Karkonoska Państwowa Szkoła Wyższa w Jeleniej Górze Wydział Nauk Medycznych i Technicznych, Katedra Nauk Informatyczno-Technicznych				
Prowadzący przedmiot:						
I. Formy zajęć, liczba godzin						
Semestr	Wykład	Ćwicz.	Laboratorium	Projekt	Seminarium	Łącznie
III	15		30			45
Forma zaliczenia	Zaliczenie na ocenę		Zaliczenie na ocenę			
Liczba punktów ECTS	1		2			3
II. Cel przedmiotu:						
C1	Uzyskanie wiedzy z zakresu technik projektowania i programowania zorientowanego obiektowo, ze szczególnym naciskiem na projektowanie i programowanie aplikacji z zastosowaniem GUI.					
C2	Nabycie umiejętności samodzielnego pisania prostych programów oraz programów o znacznym stopniu komplikacji, zgodnie z zasadami eleganckiego pisania programów i zasadami etyki.					
III. Wymagania wstępne w zakresie wiedzy, umiejętności i innych kompetencji:						
Zaliczenie przedmiotu „Podstawy programowania”.						
IV. Oczekiwane efekty uczenia się:						
Wiedza						
EK1	Ma uporządkowaną wiedzę ogólną na temat zasad programowania obiektowego w języku obiektowym (np. Java) oraz wiedzę szczegółową na temat wyrażeń i instrukcji sterujących programem, operacji wejścia/wyjścia, strumieni, kontenerów obiektów, obsługi wyjątków i wiedzę podstawową w zakresie GUI, jak również metodologii tworzenia projektów, których znajomość wykorzysta w tworzeniu aplikacji zorientowanych obiektowo.					
Umiejętności						
EK2	Ma niezbędne umiejętności konfiguracji środowiska programistycznego oraz potrafi zastosować w praktyce wiedzę szczegółową dotyczącą zasad i technik programowania obiektowego podczas implementacji prostych zadań programistycznych w języku zorientowanym obiektowo z zachowaniem zasad eleganckiego pisania programów poprzez zastosowanie komentarzy, wcięć i optymalizacji.					

EK3	Posiada umiejętność czytania i analizowania kodu programów w języku obiektowym.	
EK4	Posiada umiejętności weryfikacji wykonanych rozwiązań oraz radzenia sobie z typowymi błędami programistycznymi.	
Kompetencje społeczne		
EK5	Rozumie potrzebę samodzielnego uzupełniania i doskonalenia wiedzy oraz umiejętności związanych z programowaniem obiektowym oraz potrzebę systematycznej pracy własnej.	
EK6	Przestrzega zasad etyki i ochrony własności intelektualnej.	
EK7	Wykazuje aktywną postawę i chęć współpracy z innymi podczas rozwiązywania trudnych zadań.	
V. Treści programowe:		
Forma zajęć: wykład		Liczba godzin
Wyk1	<b>Zajęcia organizacyjne:</b> Wprowadzenie do programowania obiektowego. Paradygmaty programowania obiektowego.	1
Wyk2	<b>Podstawowe pojęcia:</b> Klasa, obiekt, pole, metoda, dziedziczenie, polimorfizm, enkapsulacja.	1
Wyk3	<b>Wstęp do języka Java:</b> Wyrażenia i instrukcje sterujące programem. Porównanie z innymi językami programowania.	1
Wyk4	<b>Zasady programowania obiektowego w Javie:</b> Ochrona danych, kapsułkowanie. Wielokrotne wykorzystanie klas. Kontrola dostępu, modyfikatory dostępu. Przykłady zastosowań.	1
Wyk5	<b>Dziedziczenie i agregacja:</b> Agregacja. Klasa bazowa i klasa pochodna. Dziedziczenie jednobazowe a dziedziczenie wielobazowe. Różnica między agregacją i dziedziczeniem. Hierarchizacja. Zalety i wady dziedziczenia. Przykłady.	1
Wyk6	<b>Polimorfizm:</b> Przesłanianie metod. Przeciążanie metod. Różnice między przesłanianiem i przeciążaniem. Przeciążanie konstruktorów. Wirtualizacja operacji. Przykłady.	1
Wyk7	<b>Interfejsy:</b> Definicja interfejsu. Proste przykłady. Klasy i metody abstrakcyjne. Dziedziczenie interfejsów. Polimorfizm przez interfejs.	1
Wyk8	<b>Kontenery obiektów:</b> Tablice i kolekcje. Kolekcje jako interfejsy, klasy abstrakcyjne i klasy konkretne. Hierarchia interfejsów kolekcji. Omówienie podstawowych interfejsów kolekcji. Hierarchia kontenerów. Omówienie najpopularniejszych kontenerów.	1
Wyk9	<b>Obsługa wyjątków w Javie:</b> Sytuacje wyjątkowe. Mechanizm obsługi wyjątków. Hierarchia wyjątków w Javie. Przykłady.	1
Wyk10	<b>Operacje wejścia-wyjścia. Strumienie w Javie:</b> System wejścia/wyjścia. Strumienie. Standardowe wejście/wyjście. Schemat przetwarzania i obsługa plików w Javie. Przykłady.	1
Wyk11	<b>Graficzny Interfejs Użytkownika:</b> Różne możliwości obsługi GUI. AWT, biblioteka SWING, SWT. Omówienie podstawowych komponentów biblioteki SWING i SWT. Podobieństwa i różnice. Przykłady zastosowań.	2
Wyk12	<b>UML:</b> Wzorce projektowe. Wprowadzenie do UML i do metodologii tworzenia projektów.	2
Wykl12	<b>Zaliczenie wkładów</b>	1
Suma godzin		30
Forma zajęć: laboratorium		Liczba godzin
Lab1	<b>Środowisko uruchomieniowe Javy:</b> Środowisko Eclipse, tworzenie projektu, szablony, kompilowanie, korzystanie z debugera).	2

Lab2	<b>Podstawy tworzenia i używania klas:</b> Tworzenie klas i obiektów tych klas. Pola i metody. Implementacja metod.	4
Lab3	<b>Konstruktor:</b> Tworzenie własnych konstruktorów. Korzystanie z konstruktorów.	2
Lab4	<b>Dziedziczenie:</b> Posługiwanie się dziedziczeniem i hierarchizacją. Różnica między dziedziczeniem i agregacją. Tworzenie metod dziedziczących.	2
Lab5	<b>Polimorfizm:</b> Przesłanianie i przeciążanie metod. Przeciążanie konstruktorów.	2
Lab6	<b>Interfejsy:</b> Różnica między interfejsami i klasami. Odczytywanie kodu źródłowego programu z interfejsami. Implementacja własnych interfejsów.	4
Lab7	<b>Kontenery:</b> Wykorzystanie kontenerów różnego rodzaju do przechowywania informacji.	4
Lab8	<b>Obsługa wyjątków:</b> Obsługa wyjątków programowych. Tworzenie własnych wyjątków i ich obsługa.	2
Lab9	<b>Strumienie:</b> Różne sposoby przetwarzania plików w Javie.	2
Lab10	<b>GUI:</b> Tworzenie prostego projektu z wykorzystaniem GUI.	4
Lab11	<b>Zaliczenie laboratorium</b>	2
<b>Suma godzin - laboratorium</b>		<b>30</b>
<b>VI. Narzędzia dydaktyczne:</b>		
1.	Prezentacje multimedialne z wykładu.	
2.	Listy ćwiczeń laboratoryjnych.	
3.	Oprogramowanie - środowisko Eclipse.	
4.	Zadania testowe.	
5.	System e-learning – publikowanie materiałów dydaktycznych (prezentacji z wykładów, list laboratoryjnych) i ogłoszeń, gromadzenie prac studenckich z laboratoriów, udostępnianie dodatkowych materiałów.	
<b>VII. Sposoby oceny (F – formująca, P – podsumowująca)</b>		
F1	Ocena punktowa rozwiązań list laboratoryjnych. Listy zadań to zbiory różnych zadań, zazwyczaj możliwych do rozwiązania podczas zajęć obejmujących od dwóch do czterech godzin lekcyjnych. Za rozwiązanie każdej listy zadań student otrzymuje punkty zależne od jakości, samodzielności i tempa wykonanej pracy. Do zaliczenia zadania konieczne jest uzyskanie co najmniej 50%punktów.	
P1	Ocena końcowa z wykładu ustalana jest na podstawie aktywności i pisemnego kolokwium zaliczeniowego	
P2	Ocena końcowa z laboratorium wyznaczana jest na podstawie sumy punktów uzyskanych przez studenta ze wszystkich list laboratoryjnych (F1). Ocena pozytywna przyznawana jest studentowi, który zdobył łącznie przynajmniej 50% sumy wszystkich punktów możliwych do uzyskania w ramach oceny F1.	
<b>VIII. Obciążenie pracą studenta</b>		
<b>Forma aktywności</b>		<b>Łączna i średnia liczba godzin na zrealizowanie aktywności</b>
Godziny kontaktowe z nauczycielem (w trakcie zajęć)		<b>60</b>

Przygotowanie się do zajęć laboratoryjnych (średnio na studenta)	<b>30</b>
Przygotowanie się do egzaminu	<b>30</b>
Konsultacje	<b>20</b>
<b>SUMA</b>	<b>140</b>
<b>SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU</b>	<b>5</b>

#### **IX. Literatura podstawowa i uzupełniająca**

##### **Literatura podstawowa:**

1. Cornell G., Horstmann C. *Java 2. Podstawy*. Wydawnictwo HELION, Gliwice 2003.
2. Eckel B. *Thinking in Java. Edycja polska*. Wydanie IV, Wydawnictwo HELION, Gliwice 2006.

##### **Literatura uzupełniająca:**

1. Cornell G., Horstmann C. *Java 2. Techniki zaawansowane. Wydanie II*. Wydawnictwo HELION, Gliwice 2005.

#### **X. Metody dydaktyczne**

<b>M1</b>	Wykład z elementami dyskusji.
<b>M2</b>	Dyskusja nad prezentowanymi przykładowymi rozwiązaniami.
<b>M3</b>	Demonstracje przykładowych rozwiązań zadań.
<b>M4</b>	Ćwiczenia symulacyjne.
<b>M5</b>	Indywidualne konsultacje podczas zajęć
<b>M6</b>	Zajęcia laboratoryjne – lista zadań wykonywana przez studentów zgodnie z poleceniami, z bieżącym wspomaganie prowadzącego

#### **XI. Tablica powiązań efektów przedmiotowych i kierunkowych z celami przedmiotu oraz stosowanymi metodami dydaktycznymi**

<b>Efekty Uczenia się</b>	<b>Odniesienie danego efektu do efektów zdefiniowanych dla programu kierunku</b>	<b>Cele przedmiotu</b>	<b>Treści programowe</b>	<b>Narzędzia dydaktyczne</b>	<b>Metody dydaktyczne</b>
<b>EK1</b>	K_W12, K_W13	C1, C2	Wyk1 – Wyk12	1, 3, 4	M1, M2, M3
<b>EK2</b>	K_U12, K_U13	C1, C2	Lab1 – Lab10	2, 3, 4, 5	M2, M3, M4, M5, M6
<b>EK3</b>	K_U12, K_U13	C1, C2	Lab1 – Lab10,	2, 3, 4, 5	M1, M2, M3, M4, M5, M6
<b>EK4</b>	K_U12, K_U13	C1, C2	Lab1 – Lab10	2, 3, 4, 5	M2, M3, M4, M5, M6

EK5	K_K01,K_K06	C1, C2	Lab1 – Lab10, Wyk1 - Wyk12	1, 2, 3, 4, 5	M5, M6	
EK6	K_K04, K_K06	C1, C2	Lab1 – Lab10, Wyk1 - Wyk12	1, 2, 3, 4, 5	M2, M3, M5, M6	
XII. Zasady weryfikacji oczekiwanych efektów uczenia się						
Efekt uczenia się	Sposób weryfikacji					
EK1	P1					
EK2	F1, P2					
EK3	F1, P2					
EK4	F1, P2					
Kryteria weryfikacji						
Sposób weryfikacji	Na ocenę 2.0	Na ocenę 3.0	Na ocenę 3.5	Na ocenę 4.0	Na ocenę 4.5	Na ocenę 5.0
P1	Suma punktów uzyskanych z list zadań laboratoryjnych (ocen F1) jest mniejsza niż 50 % sumy wszystkich punktów możliwych do uzyskania w ramach ocen F1.	Suma punktów uzyskanych z list zadań laboratoryjnych (ocen F1) jest nie mniejsza niż 50 % sumy wszystkich punktów możliwych do uzyskania w ramach ocen F1.	Suma punktów uzyskanych z list zadań laboratoryjnych (ocen F1) jest mniejsza niż 60 % sumy wszystkich punktów możliwych do uzyskania w ramach ocen F1.	Suma punktów uzyskanych z list zadań laboratoryjnych (ocen F1) jest mniejsza niż 70 % sumy wszystkich punktów możliwych do uzyskania w ramach ocen F1.	Suma punktów uzyskanych z list zadań laboratoryjnych (ocen F1) jest mniejsza niż 80 % sumy wszystkich punktów możliwych do uzyskania w ramach ocen F1.	Suma punktów uzyskanych z list zadań laboratoryjnych (ocen F1) jest mniejsza niż 90 % sumy wszystkich punktów możliwych do uzyskania w ramach ocen F1.
P2	Suma punktów uzyskanych z kolokwium zaliczeniowego jest mniejsza niż 50 % sumy wszystkich punktów możliwych do uzyskania.	Suma punktów uzyskanych z kolokwium zaliczeniowego jest nie mniejsza niż 50 % sumy wszystkich punktów możliwych do uzyskania	Suma punktów uzyskanych z kolokwium zaliczeniowego wynosi co najmniej 60 % sumy wszystkich punktów możliwych do uzyskania	Suma punktów uzyskanych z kolokwium zaliczeniowego wynosi co najmniej 70 % sumy wszystkich punktów możliwych do uzyskania	Suma punktów uzyskanych z kolokwium zaliczeniowego wynosi co najmniej 80 % sumy wszystkich punktów możliwych do uzyskania F2.	Suma punktów uzyskanych z kolokwium zaliczeniowego wynosi co najmniej 90 % sumy wszystkich punktów możliwych do uzyskania F2.
XII.NNE PRZYDATNE INFORMACJE O PRZEDMIOCIE						