

KARTA PRZEDMIOTU**I. Dane podstawowe**

Nazwa przedmiotu	Programowanie obiektowe
Nazwa przedmiotu w języku angielskim	Object-oriented programming
Kierunek studiów	Informatyka
Poziom studiów (I, II, jednolite magisterskie)	I-go stopnia
Forma studiów (stacjonarne, niestacjonarne)	Stacjonarne
Dyscyplina	informatyka, matematyka
Język wykładowy	język polski

Koordinator przedmiotu/osoba odpowiedzialna	dr Henryk Malinowski
---	----------------------

Forma zajęć (<i>katalog zamknięty ze słownika</i>)	Liczba godzin	semestr	Punkty ECTS
wykład	30	III	5
konwersatorium			
ćwiczenia			
laboratorium	30	III	
warsztaty			
seminarium			
proseminarium			
lektorat			
praktyki			
zajęcia terenowe			
pracownia dyplomowa			
translatorium			
wizyta studyjna			

Wymagania wstępne	Znajomość podstaw algorytmiki i programowania. Znajomość podstaw języka C++.
-------------------	---

II. Cele kształcenia dla przedmiotu

C1. Zapoznanie z metodologią i techniką programowania obiektowego.
C2. Poszerzenie znajomości języka C++.

III. Efekty uczenia się dla przedmiotu wraz z odniesieniem do efektów kierunkowych

Symbol	Opis efektu przedmiotowego	Odniesienie do efektu kierunkowego
WIEDZA		
W_01	Student potrafi przedstawić podstawowe pojęcia programowania obiektowego	K_W01, K_W06
W_02	Student potrafi analizować pliki źródłowe aplikacji obiektowych	K_W03, K_W06;
UMIEJĘTNOŚCI		
U_01	Student potrafi stosować zasady definiowania klas, tworzenia obiektów i modelować wybrane zagadnienia w sposób obiektowy	K_U04, K_U06, K_U08, K_U11, K_U17
U_02	Student potrafi napisać prostą aplikację w obiektowym języku programowania	K_U07, K_U08, K_U10, K_U11, K_U12
U_03	Student potrafi stosować dziedziczenie i polimorfizm oraz przeciążać operatory	K_U10, K_U11
KOMPETENCJE SPOŁECZNE		
K_01	Student potrafi pracować indywidualnie oraz z innymi osobami nad wybranym problemem	K_K01, K_K04, K_K07

IV. Opis przedmiotu/ treści programowe

Klasy. Pola i metody. Dostęp do składowych klasy.
 Deklarowanie obiektów klasy, konstruktory, destruktor.
 Wskaźniki i obiekty dynamiczne. Wskaźnik this.
 Obsługa błędów za pomocą wyjątków.
 Mechanizm zaprzyjaźniania.
 Przeciążanie operatorów.
 Dziedziczenie.
 Kolejność wywoływania konstruktorów i destruktorów.
 Polimorfizm i metody wirtualne.
 Dziedziczenie wielokrotne i wirtualne.
 Klasy abstrakcyjne.
 Pola i metody statyczne.
 Strumieniowe operacje wejścia wyjścia.

V. Metody realizacji i weryfikacji efektów uczenia się

Symbol efektu	Metody dydaktyczne (lista wyboru)	Metody weryfikacji (lista wyboru)	Sposoby dokumentacji (lista wyboru)
WIEDZA			
W_01	Wykład konwencjonalny, Praca pod kierunkiem	Egzamin/Kolokwium	Sprawdzian pisemny / Uzupełnione i ocenione kolokwium
W_02	Wykład konwencjonalny, Praca pod kierunkiem	Egzamin/Kolokwium	Sprawdzian pisemny / Uzupełnione i ocenione kolokwium
UMIEJĘTNOŚCI			
U_01	Ćwiczenia praktyczne	Egzamin/Kolokwium	Sprawdzian pisemny / Uzupełnione i ocenione kolokwium
U_02	Ćwiczenia praktyczne	Egzamin/Kolokwium	Sprawdzian pisemny / Uzupełnione i ocenione kolokwium
U_03	Ćwiczenia praktyczne	Egzamin/Kolokwium	Sprawdzian pisemny / Uzupełnione i ocenione kolokwium
KOMPETENCJE SPOŁECZNE			
K_01	Dyskusja	Egzamin/Kolokwium	Sprawdzian pisemny / Uzupełnione i ocenione kolokwium

Kryteria oceny, wagi...

... Zaliczenie ćwiczeń: weryfikacja za pomocą pisemnych trzech sprawdzianów pisemnych na 5., 9. i 13. ćwiczeniach (20% oceny końcowej), aktywności i odpowiedzi ustnych na laboratoriach (10% oceny końcowej), prac domowych obejmujących wybrane tematy programowania obiektowego (20% oceny końcowej) oraz kolokwium (50% oceny końcowej). Sprawdzian pisemny może być przesunięty na inny termin po uzgodnieniu ze studentami.

Zaliczenie wykładu: egzamin pisemny dla osób, które zaliczyły ćwiczenia.

Kryteria oceny: poniżej 50% wyniku końcowego – ocena niedostateczna.

Szczegółowe zasady oceniania są podawane studentom z każdą edycją przedmiotu.

VI. Obciążenie pracą studenta

Forma aktywności studenta	Liczba godzin
Liczba godzin kontaktowych z nauczycielem	90
Liczba godzin indywidualnej pracy studenta	60

VII. Literatura

Literatura podstawowa
<ol style="list-style-type: none">1. B.Meyer, Programowanie zorientowane obiektowo, Helion, 20052. I. Graham, Metody obiektowe w teorii i praktyce, WNT Warszawa, 20043. H.M. Deitel, P. J. Deitel, ARKANA C++ Programowanie, Wyd. RM, Warszawa, 1998.4. J. Grębosz, Opus Magnum C++11 : programowanie w języku C++. T. 1, Helion, 2018
Literatura uzupełniająca
<ol style="list-style-type: none">1. Grębosz J., Symfonia C++ standard, Wyd. Edition 2000. Kraków, 2006.2. B. Eckel, Thinking in C++, Helion, Gliwice 2002.3. N. Dale, Ch. Weems, M. Headington, Programming and problem solving with C++. Jones and Bartlett Publishers, Boston, 2002.4. Zasoby internetowe