


KAPITAŁ LUDZKI
 NARODOWA STRATEGIA SPÓJNOŚCI

 Projekt współfinansowany przez
 Unię Europejską w ramach
 Europejskiego Funduszu
 Społecznego

UNIA EUROPEJSKA
 EUROPEJSKI
 FUNDUSZ SPOŁECZNY


Nazwa przedmiotu		Kod ECTS	
Zaawansowane programowanie w C++ (MSU)		11.0.0260	
Nazwa jednostki prowadzącej przedmiot			
Instytut Informatyki			
Studia			
wydział	kierunek	poziom	drugiego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
dr Tomasz Borzyszkowski; mgr inż. Kamil Legierski			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		3 Udział w zajęciach dydaktycznych, objętych planem studiów: 30h Praca własna studenta: 45h RAZEM: 75h	
Wykład, Ćw. laboratoryjne			
Sposób realizacji zajęć			
zajęcia on-line, zajęcia w sali dydaktycznej			
Liczba godzin			
Wykład: 15 godz., Ćw. laboratoryjne: 15 godz.			
Termin realizacji przedmiotu			
2023/2024 letni			
Status przedmiotu		Język wykładowy	
fakultatywny (do wyboru)		polski	
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne	
<ul style="list-style-type: none"> - Wykład z prezentacją multimedialną - ćwiczenia laboratoryjne - przygotowanie programów związanych z kryptografią, zapoznanie się z istniejącymi rozwiązaniami 		Sposób zaliczenia	
		Zaliczenie na ocenę	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru - kolokwium 	
		Podstawowe kryteria oceny	
		Ocena z laboratorium: <ul style="list-style-type: none"> • 100%: ocena projektów studentów Ocena z wykładu: <ul style="list-style-type: none"> • na podstawie pracy na części laboratoryjnej 	
Sposób weryfikacji założonych efektów uczenia się			

zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność w dyskusji	obserwacja postawy
Wiedza							
K_W05			X			X	
P_W1			X				
Umiejętności							
K_U07			X				
K_U10			X			X	
P_U1			X				
Kompetencje							
K_K01							X
K_K03						X	
P_K1			X				X
P_K2							X

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi**A. Wymagania formalne**

Przedmioty wprowadzające: Języki programowania, Bazy danych, Programowanie obiektowe.

B. Wymagania wstępne

Znajomość podstaw programowania obiektowego oraz relacyjnych baz danych.

Cele kształcenia

Nauczenie studentów podstawowych konstrukcji programistycznych dostępnych w języku C++.

Treści programowe

1. Podstawy programowania obiektowego (konstruktory, destruktory, zarządzanie pamięcią)
2. Dziedziczenie i polimorfizm
3. Przeciążanie operatorów i wyjątki
4. Biblioteka STL
5. Semantyka move i wyrażenia lambda
6. Inteligentne wskaźniki
7. Szablony i koncepcje

Wykaz literatury

1. B. Stroustrup. Język C++. Kompendium wiedzy, Helion 2014
2. N. Josuttis "C++ Biblioteka Standardowa, podręcznik programisty", Wydawnictwo HELION 2003.
3. B. Eckel, Ch. Allinson, "Thinking in C++" Edycja Polska, tom 2, Wydawnictwo HELION 2004.
4. C++ Reference, <http://en.cppreference.com/w/>
5. <http://www.cplusplus.com/>

Kierunkowe efekty uczenia się

K_W05: ma ogólną wiedzę na temat różnych paradygmatów programowania i języków programowania; szczególnie zna metody i wzorce projektowania i programowania obiektowego

K_U07: potrafi projektować, tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz adekwatnych wzorców

K_U10: potrafi oceniać przydatność paradygmatów i narzędzi programistycznych do rozwiązywania problemów różnego typu

K_K01: zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się

K_K03: potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych

Wiedza

P_W1: Student rozumie zasady działania skryptowych języków programowania i wie jak działają w tych językach podstawowe konstrukcje programistyczne (K_W05)

Umiejętności

P_U1: Student umie zastosować wybrane biblioteki języka C++ do rozwiązania problemów programistycznych (K_U07, K_U10)

Kompetencje społeczne (postawy)

P_K1: Student zdaje sobie sprawę z tempa rozwoju technologicznego i rozumie potrzebę dalszego uczenia się (K_K01)

P_K2: Student potrafi formułować opinie na temat zastosowania podstawowych konstrukcji programistycznych (K_K03)

Kontakt

tb@inf.ug.edu.pl