

KARTA PRZEDMIOTU

Nazwa w języku polskim: **Podstawy programowania 2**

Nazwa w języku angielskim: **Programming Basics part 2**

Karta przedmiotu ważna od roku akademickiego: **2022/2023**

Kierunek studiów: **Informatyka**

Poziom studiów: **Studia I stopnia**

Forma studiów: **Niestacjonarne**

Profil: **Praktyczny**

Specjalność: **Cyberbezpieczeństwo, Systemy i sieci komputerowe, Informatyka przemysłowa**

Język wykładowy: **Polski**

Jednostka prowadząca: **Wydział Nauk Społecznych i Technicznych**

Prowadzący: **dr Grzegorz Jastrzębski**

OBCIĄŻENIE STUDENTA

	Wykład	Konwersatorium	Laboratorium	Projekt	Seminarium
Liczba godzin zajęć dydaktycznych organizowanych przez Uczelnię	14		16		
Liczba godzin całkowitego nakładu pracy studenta	25		50		
Forma zaliczenia	Egzamin		Zaliczenie z oceną		
Liczba punktów ECTS	1		2		

WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

Wiedza i umiejętności z zakresu projektowania efektywnych algorytmów przetwarzających.

Umiejętność przeczytania, interpretacji oraz symbolicznego wykonania prostych programów napisanych w języku programowania imperatywnego.

Umiejętność posługiwania się zintegrowanym środowiskiem programistycznym w celu edycji i uruchomienia napisanego przez siebie prostego programu.

CELE PRZEDMIOTU

C1	Nabycie wiedzy i umiejętności w zakresie metod modularyzacji kodu właściwych dla współczesnych języków programowania proceduralnego.
C2	Nabycie wiedzy i umiejętności w zakresie metod współdzielenia danych między różnymi modułami programu.
C3	Nabycie wiedzy i umiejętności w zakresie rekurencyjnej implementacji algorytmów w programowaniu proceduralnym.

PRZEDMIOTOWE EFEKTY UCZENIA – PEU	
Z zakresu wiedzy:	
PEU_W01	Student zna techniki modularyzacji kodu stosowane w programowaniu proceduralnym.
PEU_W02	Student zna iteracyjne oraz rekurencyjne metody implementacji algorytmów.
Z zakresu umiejętności:	
PEU_U01	Student potrafi stosować techniki przekazywania i współdzielenia danych pomiędzy modułami programu.
PEU_U02	Student potrafi opracować rekurencyjne implementacje wybranych algorytmów.
Z zakresu kompetencji społecznych:	
PEU_K01	Student jest kreatywny w wyborze środowiska do budowy informatycznego systemu użytkowego.

TREŚCI PROGRAMOWE		
Forma zajęć – wykład		Liczba godzin
W1	Wprowadzenie, omówienie zasad zaliczenia przedmiotu. Cele i techniki modularyzacji kodu.	4
W2	Mechanizmy przekazywania parametrów i zwracania wartości podprogramów.	4
W3	Kategorie zmiennych i czas ich życia.	3
W4	Rekurencja jako alternatywna implementacja algorytmów iteracyjnych.	3
Razem		14

TREŚCI PROGRAMOWE		
Forma zajęć – laboratoria		Liczba godzin
L1	Wprowadzenie, omówienie zasad zaliczenia przedmiotu. Zintegrowane środowisko rozwijania oprogramowania (IDE) – podstawowe procedury edycji i uruchamiania programów.	2
L2	Tworzenie podprogramów realizujących zasadę separacji kodu.	6
L3	Użycie różnych kategorii zmiennych (globalnych, lokalnych, lokalnych statycznych) do przekazywania informacji między podprogramami.	6
L4	Użycie typowych schematów przetwarzania rekurencyjnego w implementacjach algorytmów. Podsumowanie, zaliczenie.	2
Razem		16

STOSOWANE NARZĘDZIA DYDAKTYCZNE

1.	Wykład. Prezentacja treści z wykorzystaniem multimediiów.
2.	Laboratorium z wykorzystaniem stanowiska laboratoryjnego i oprogramowania komputerowego.
3.	Dyskusja dydaktyczna.
4.	Praca własna – studiowanie aktualnej literatury przedmiotu, źródeł internetowych.

METODY I FORMY OCENY

OSIĄGNIĘCIA PRZEDMIOTOWYCH EFEKTÓW UCZENIA

Formy oceny (F lub P)*	Numer efektu uczenia (przedmiotowego)	Metody oceny osiągnięcia efektu uczenia
F I	PEU_U01, PEU_U02, PEU_K01	Ocena za aktywną realizację ćwiczeń laboratoryjnych.
P I	PEU_U01, PEU_U02, PEU_K01	Ocena wykonania zadań na zajęciach zaliczeniowych.
F w	PEU_W01, PEU_W02, PEU_K01	Dyskusja w trakcie zajęć.
P w	PEU_W01, PEU_W02, PEU_K01	Egzamin w formie pisemnej.

*F – ocena formująca (w trakcie semestru), P – ocena podsumowująca (na koniec semestru)

KRYTERIA OCENY

OSIĄGNIĘCIA PRZEDMIOTOWYCH EFEKTÓW UCZENIA

Nr PEU	Ocena dostateczna	Ocena dobra	Ocena bardzo dobra
PEU_W01	Student zna podstawowe techniki modularyzacji kodu stosowane w programowaniu proceduralnym.	Student zna techniki modularyzacji kodu stosowane w programowaniu proceduralnym oraz potrafi je stosować w uruchamianych samodzielnie programach.	Student zna techniki modularyzacji kodu stosowane w programowaniu proceduralnym oraz potrafi dokonać ich krytycznej analizy oraz właściwego doboru w uruchamianych samodzielnie programach.
PEU_W02	Student zna podstawowe przykłady iteracyjnych oraz rekurencyjnych metod implementacji algorytmów.	Student zna iteracyjne oraz rekurencyjne metody implementacji algorytmów oraz ma wiedzę w zakresie ich stosowania przy samodzielnej pracy nad projektem programu.	Student zna iteracyjne oraz rekurencyjne metody implementacji algorytmów oraz potrafi dokonać ich krytycznej analizy oraz właściwego doboru przy samodzielnej pracy nad projektem programu.
PEU_U01	Student potrafi przekazywać i współdzielić dane pomiędzy modułami prostego programu wzorując się na rozwiązaniach standardowych.	Student potrafi samodzielnie zastosować w praktyce techniki przekazywania i współdzielenia danych pomiędzy modułami programu.	Student potrafi samodzielnie dokonać krytycznej analizy oraz optymalnego doboru technik przekazywania i współdzielenia danych pomiędzy modułami uruchamianego programu.
PEU_U02	Student potrafi zaproponować rekurencyjne implementacje wybranych standardowych.	Student potrafi samodzielnie zastosować rekurencyjne implementacje wybranych algorytmów w uruchamianych	Student potrafi samodzielnie dokonać krytycznej analizy oraz optymalnego doboru rekurencyjnych implementacji

	Algorytmów.	przez siebie programach.	wybranych algorytmów w uruchamianych przez siebie programach.
PEU_K01	Student jest gotów kreatywnie dokonać wyboru środowiska do budowy informatycznego systemu użytkowego.	Student jest kreatywny w wyborze różnych środowisk do budowy informatycznego systemu użytkowego.	Student jest kreatywny w wyborze różnych środowisk do budowy złożonego informatycznego systemu użytkowego.

LITERATURA PODSTAWOWA

Grębosz J., Symfonia C++ standard, Wyd. Editions 2000, Kraków, 2008.
Neapolitan R., Podstawy algorytmów z przykładami w C++, Gliwice Helion, 2004.

LITERATURA UZUPEŁNIAJĄCA

Stroustrup B., Język C++, WNT, Warszawa, 2002.
Koza. Z., Język C++. Pierwsze starcie, Helion, 2008.

MACIERZ POWIĄZANIA EFEKTÓW UCZENIA DLA PRZEDMIOTU PODSTAWY PROGRAMOWANIA 2 Z EFEKTAMI UCZENIA NA KIERUNKU INFORMATYKA

Przedmiotowy efekt uczenia	Odniesienie przedmiotowego efektu do efektów uczenia zdefiniowanych dla kierunku studiów i specjalności	Cele przedmiotu	Treści programowe	Numer narzędzia dydaktycznego
PEU_W01	K_W03	C1	W1, L2, L3	1, 2, 4
PEU_W02	K_W03	C3	W2, W3	1, 3
PEU_U01	K_U07	C2	W2, L2, L3	1, 2, 4
PEU_U02	K_U07	C3	W4, L2, L4	1, 2, 4
PEU_K01	K_K04	C1, C2	L1	2, 3