

POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2021/2022

Wydział Inżynierii Elektrycznej i Komputerowej

Kierunek studiów: Informatyka w Inżynierii Komputerowej

Profil: Ogólnoakademicki

Forma studiów: stacjonarne

Kod kierunku: IwIK

Stopień studiów: I

Specjalności: bez specjalności

1 INFORMACJE O PRZEDMIOCIE

NAZWA PRZEDMIOTU	Metody programowania
NAZWA PRZEDMIOTU W JĘZYKU ANGIELSKIM	Programming Methods
KOD PRZEDMIOTU	WIEiK INFOR_W_INZ_KOMP oIS PK6 21/22
KATEGORIA PRZEDMIOTU	Przedmioty kierunkowe
LICZBA PUNKTÓW ECTS	6.00
SEMESTRY	2

2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

SEMESTR	WYKŁADY	ĆWICZENIA	LABORATORIA	LABORATORIA KOMPUTERO- WE	PROJEKTY	
2	45	0	0	45	0	0

3 CELE PRZEDMIOTU

Cel 1 Poznanie podstawowych pojęć i zagadnień związanych z programowaniem i złożonością obliczeniową algorytmów.

Cel 2 Ugruntowanie znajomości podstawowych struktur danych i ich zastosowań. Poznanie wybranych zaawansowanych struktur danych i ich zastosowań.

- Cel 3** Znajomość zbioru klasycznych metod programowania, zbioru wybranych zaawansowanych metod programowania, metod generacji obiektów kombinatorycznych i liczb losowych oraz obszarów zastosowań każdej z nich.
- Cel 4** Napisanie i uruchomienie programów implementujących algorytmy klasycznych metod programowania i wybranych zaawansowanych metod programowania, programowe generatory obiektów kombinatorycznych i generatory liczb losowych. Udoskonalenie umiejętności programowania w języku C/C++ oraz indywidualnej pracy programistycznej.
- Cel 5** Wykonanie wybranego zespołowego projektu programistycznego. Udoskonalenie umiejętności programowania w języku C/C++ oraz zespołowej pracy programistycznej.

4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

- 1 Zaliczony przedmiot z pierwszego semestru: Wstęp do programowania.

5 EFEKTY KSZTAŁCENIA

- EK1 Wiedza** Znajomość wybranych zagadnień związanych z programowaniem (dualizm programowo-sprzętowy, hipoteza Sapira-Whorfa, klasyfikacja problemów, notacje asymptotyczne, złożoność czasowa i pamięciowa, hierarchia funkcji złożoności obliczeniowej, teorie złożoności obliczeniowej, klasyfikacja algorytmów, programowanie strukturalne, programowanie obiektowe, programowanie wizualne, styl programowania, przenośność oprogramowania).
- EK2 Wiedza** Znajomość podstawowych i wybranych zaawansowanych struktur danych (tablica, lista, kolejki FIFO i LIFO (stos), kolejka priorytetowa, graf nieskierowany i skierowany, drzewo, kopiec), ich zastosowań oraz związku pomiędzy wyborem struktury danych a złożonością algorytmu.
- EK3 Wiedza** Znajomość poznanego zbioru klasycznych metod programowania (rekursja, programowanie dynamiczne, wyszukiwanie wyczerpujące, wyszukiwanie z powrotami, algorytmy zachłanne, wyszukiwanie lokalne) i obszaru zastosowania każdej z nich.
- EK4 Wiedza** Znajomość podstaw wybranych zaawansowanych metod programowania (metody iteracyjne, metody równoległe, metody geometrii obliczeniowej) i obszaru zastosowania każdej z nich.
- EK5 Wiedza** Znajomość reprezentacji i metod generowania różnych klas obiektów kombinatorycznych, metod programowej i sprzętowej generacji liczb losowych wraz z ich obszarami zastosowań.
- EK6 Umiejętności** Napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru klasycznych metod programowania
- EK7 Umiejętności** Napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru zaawansowanych metod programowania.
- EK8 Umiejętności** Napisanie i uruchomienie programów generujących obiekty kombinatoryczne i liczby losowe.
- EK9 Kompetencje społeczne** Praca w małym zespole, podział zadań, efektywna współpraca w osiągnięciu wyznaczonego celu, dzielenie się wiedzą, wywiązywanie się z przyjętych obowiązków, kierowanie pracą zespołu.

6 TREŚCI PROGRAMOWE

WYKŁADY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BŁOKÓW TEMATYCZNYCH	LICZBA GODZIN

WYKŁADY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W1	Definicja programowania. Rys historyczny rozwoju programowania. Dualizm programowo-sprzętowy. Języki i środowiska programowania. Hipoteza Sapira-Whorfa. Programowanie - klasyfikacje. Wymagania jakościowe. Wprowadzenie do algorytmów i ich złożoności obliczeniowej. Klasyfikacja problemów. Notacje asymptotyczne. Złożoność czasowa i pamięciowa. Klasyfikacja algorytmów. Hierarchia funkcji złożoności. Obszary zastosowań algorytmiki. Metody programowania.	3
W2	Podstawowe struktury danych, ich reprezentacje, elementarne operacje i przykłady zastosowań. Tablica. Lista i jej warianty: kolejka FIFO, kolejka LIFO (stos), kolejka podwójna, kolejka priorytetowa. Zbiór. Graf nieskierowany i skierowany. Drzewo. Kopiec. Struktury, unie. Abstrakcyjne typy danych.	3
W3	Klasyczne techniki programowania. Programowanie strukturalne. Instrukcje strukturalne. Metody projektowania programów. Zasada abstrakcji - poziomy abstrakcji. Programowanie modularne. Graficzne przedstawianie programów. Schematy blokowe i diagramy strukturalne. Niestrukturalność. Programowanie obiektowe - charakterystyka i podstawowe pojęcia. Programowanie wizualne. Automatyzacja w tworzeniu programów CASE.	3
W4	Styl programowania a jakość programu. Nazwy. Wyrażenia i instrukcje. Jednolitość stylu, idiomy. Makroinstrukcje a funkcje. Liczby magiczne. Komentarze. Rola dobrego stylu. Zasady dobrego stylu. Przenośność oprogramowania. Tworzenie przenośnego oprogramowania. Język. Pliki nagłówkowe i biblioteki. Organizacja programu. Izolowanie. Wymiana danych - kolejność bajtów. Format CVF. Przenośność i uaktualnianie. Uniwersalne kodowanie.	3
W5	Pojęcie rekursji. Rekursja liniowa i drzewiasta. Rekurencyjne wyznaczanie wartości ciągów: silnia, liczby Fibonacciego, współczynniki dwumienne. Inne algorytmy rekurencyjne: wieże Hanoi, sortowanie MergeSort i QuickSort, trawersowanie drzew binarnych, dekompozycja układów kombinacyjnych w oparciu o uogólnione twierdzenie C. Shanona o dekompozycji funkcji logicznej. Analiza zalet i wad rekursji. Eliminacja rekursji: iteracja, zastosowanie stosu. Rozwiązywanie rekurencji: metoda podstawiania, metoda iteracyjna, metoda rekurencji uniwersalnej.	3
W6	Problem generacji liczb losowych. Liczby losowe a pseudolosowe. Zastosowania generatorów liczb losowych. Metody programowej generacji liczb losowych. Przykłady obliczeniowe. Statystyczna ocena jakości generatorów. Metody sprzętowej generacji liczb losowych.	3
W7	Generacja obiektów kombinatorycznych. Podstawowe typy obiektów: kombinacje, permutacje, nieporządki, podziały, drzewa itp. Reprezentacja obiektów kombinatorycznych za pomocą funkcji wyboru rodziny indeksowanej zbiorów. Porządki leksykograficzne na zbiorze funkcji wyboru. Generacja w porządku leksykograficznym. Algorytmy generacyjne. Zastosowania : wyszukiwanie wyczerpujące, testowanie algorytmów kombinatorycznych.	3

WYKŁADY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W8	Idea programowania dynamicznego. Wyznaczanie wartości ciągów metodą programowania dynamicznego: silnia, liczby Fibonacciego, współczynniki dwumienne, liczby Stirlinga. Własność optymalnej podstruktury. Inne algorytmy programowania dynamicznego: problem podziału zbioru, dyskretny problem plecakowy, problem iloczynu łańcucha macierzy. Wymagania czasowe i pamięciowe algorytmów programowania dynamicznego.	3
W9	Idea algorytmu zachłannego. Algorytm przybliżony lub dokładny (własność optymalnej podstruktury). Elementy teorii matroidów. Klasy matroidów: podziałowe, macierzowe, grafowe, transversalne. Przykłady klasycznych algorytmów zachłannych: algorytm eliminacji Gaussa, algorytm Kruskala dla problemu MST, algorytm kodowania Huffmana, problem pokrycia zbioru, algorytmy Kaprałskiego-Skarbka dla problemu minimalnej bazy macierzy boolowskiej. Wady i zalety algorytmów zachłannych.	3
W10	Wyszukiwanie wyczerpujące w przestrzeni rozwiązań a idea algorytmu wyszukiwania z nawrotami. Przykład : Problem plecakowy, problem komiwojażera (TSP). Przykłady klasycznych algorytmów z nawrotami: problem skoczka szachowego, problem ośmiu hetmanów. Gry i ich reprezentacje w postaci drzew gier. Ocena rozwiązań w węzłach. Technika odcinania alfa-beta gałęzi drzewa gry. Wady i zalety algorytmów z nawrotami.	3
W11	Koncepcja wyszukiwania lokalnego. Przykład: problem spełnialności (SAT). Wyszukiwanie lokalne w optymalizacji. Przykłady: pokrycie wierzchołkowe grafu (VCP), minimalne drzewo rozpinające (MST), problem komiwojażera (TSP), problem minimalnej bazy macierzy boolowskiej. Przykładowe operacje i definicje sąsiedztwa. Ulepszenia i modyfikacje wyszukiwania lokalnego. Algorytm wyszukiwania z listą tabu (TS). Przykład : problem MST z ograniczeniami. Algorytm symulowanego wyżarzania (SA). Przykład: problem podziału układu logicznego. Zalety i wady wyszukiwania lokalnego.	3
W12	Idea populacyjnych algorytmów iteracyjnych. Podstawowe typy iteracyjnych algorytmów optymalizacyjnych. Optymalizacja dyskretna - problem kolorowania grafu, problem szeregowania zadań. Algorytmy ewolucyjne (EA). Algorytmy mrówkowe (ACO). Optymalizacja ciągła - optymalizacja funkcji. Algorytm roju cząstek (PSO). Schematy zrównoleglenia programów iteracyjnych. Populacyjne algorytmy hybrydowe.	3
W13	Metody geometrii obliczeniowej. Elementarne obiekty geometryczne i ich właściwości. Geograficzne systemy informacyjne. Problem przecinania się odcinków. Triangulacja wieloboku. Problem galerii sztuki. Problem znajdowania wypukłej otoczki - algorytm Jarvisa. Wyszukiwanie w zakresach - metoda siatek, metoda drzew. Problem znalezienia najkrótszej ścieżki dla robota punktowego - graf widzialności.	3

WYKŁADY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W14	Programowanie równoległe i pojęcia pokrewne. Charakterystyka systemów równoległych. Organizacja obliczeń równoległych. Przykłady obliczeń równoległych. Podstawowe miary jakości obliczeń równoległych. Zależności pomiędzy miarami jakości. Przykład oceny jakości systemu równoległego. Bariery: inherentna sekwencyjność problemów, prawa Amdahla. Modele obliczeń równoległych. Paradygmaty obliczeń równoległych: programowanie z dzieloną pamięcią, z przesyłaniem komunikatów, wielowątkowe. Zastosowania GPU w systemach równoległych. Rozwój równoległych języków programowania.	3
W15	Podstawowe pojęcia i wyniki teorii NP-zupełności, teorii P-zupełności oraz teorii aproksymowalności oraz ich konsekwencje dla algorytmiki.	3

LABORATORIA KOMPUTEROWE		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
K1	Generacja krotek, k-podzbiorów zbioru n-elementowego, podziałów zbioru, permutacji i nieporządków zbioru, drzew t-narnych o n wierzchołkach wewnętrznych - w różnych reprezentacjach i porządkach liniowych.	12
K2	Generacja grafów o zadanych własnościach (grafy losowe, grafy R-MAT).	3
K3	Proste algorytmy rekurencyjne (problem do wyboru).	3
K4	Programowanie dynamiczne (problem do wyboru).	3
K5	Algorytmy zachłanne : kodowanie Huffmana, problem minimalnej bazy macierzy boolowskiej.	3
K6	Algorytmy z powrotami, algorytm MIN-MAX i odcinanie alfa-beta (problem do wyboru).	3
K7	Wyszukiwanie lokalne : algorytmy GSAT i WALKSAT dla problemu Spełnialności (SAT), algorytm dla problemu minimalnej bazy macierzy boolowskiej.	3
K8	Algorytmy z geometrii obliczeniowej (problem do wyboru).	3
K9	Algorytmy iteracyjne, algorytmy równoległe. Projekt zespołowy (temat do wyboru).	3
K11	Laboratoria zaliczeniowe (3h) po każdych czterech tematach laboratoryjnych (12h).	9

7 NARZĘDZIA DYDAKTYCZNE

N1 Wykłady

N2 Ćwiczenia laboratoryjne

N3 Praca w grupach

8 OBCIĄŻENIE PRACĄ STUDENTA

FORMA AKTYWNOŚCI	ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI
Godziny kontaktowe z nauczycielem akademickim, w tym:	
Godziny wynikające z planu studiów	90
Konsultacje przedmiotowe	6
Egzaminy i zaliczenia w sesji	6
Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym:	
Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	20
Opracowanie wyników	12
Przygotowanie raportu, projektu, prezentacji, dyskusji	12
programowanie, testowanie programów	46
SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA	192
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	6.00

9 SPOSOBY OCENY

OCENA FORMUJĄCA

F1 Test - 10 quizów na wykładach

F2 Laboratorium : program + sprawozdanie z ćwiczenia laboratoryjnego

F3 Projekt programistyczny zespołowy : program + sprawozdanie z projektu

OCENA PODSUMOWUJĄCA

P1 Średnia ważona ocen formujących 1

P2 Średnia ważona ocen formujących 2 i 3

P3 Egzamin pisemny

WARUNKI ZALICZENIA PRZEDMIOTU

W1 Obecności na wykładach i laboratoriach komputerowych

W2 Zaliczenie laboratorium (ocena podsumowująca 2)

W3 Zdanie egzaminu (ocena podsumowująca 3)

OCENA AKTYWNOŚCI BEZ UDZIAŁU NAUCZYCIELA**B1** W ramach ocen formujących**KRYTERIA OCENY**

EFEKT KSZTAŁCENIA 1	
NA OCENĘ 2.0	Poniżej 50 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z programowaniem.
NA OCENĘ 3.0	50-59 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z programowaniem.
NA OCENĘ 3.5	60-69 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z programowaniem.
NA OCENĘ 4.0	70-79 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z programowaniem.
NA OCENĘ 4.5	80-89 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z programowaniem.
NA OCENĘ 5.0	90-100 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z programowaniem.
EFEKT KSZTAŁCENIA 2	
NA OCENĘ 2.0	Poniżej 50 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych ze strukturami danych.
NA OCENĘ 3.0	50-59 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych ze strukturami danych.
NA OCENĘ 3.5	60-69 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych ze strukturami danych.
NA OCENĘ 4.0	70-79 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych ze strukturami danych.
NA OCENĘ 4.5	80-89 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych ze strukturami danych.
NA OCENĘ 5.0	90-100 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych ze strukturami danych.
EFEKT KSZTAŁCENIA 3	
NA OCENĘ 2.0	Poniżej 50 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z klasycznymi metodami programowania
NA OCENĘ 3.0	50-59 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z klasycznymi metodami programowania

NA OCENĘ 3.5	60-69 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z klasycznymi metodami programowania
NA OCENĘ 4.0	70-79 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z klasycznymi metodami programowania
NA OCENĘ 4.5	80-89 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z klasycznymi metodami programowania
NA OCENĘ 5.0	90-100 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z klasycznymi metodami programowania
EFEKT KSZTAŁCENIA 4	
NA OCENĘ 2.0	Poniżej 50 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z zaawansowanymi metodami programowania
NA OCENĘ 3.0	50-59 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z zaawansowanymi metodami programowania
NA OCENĘ 3.5	60-69 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z zaawansowanymi metodami programowania
NA OCENĘ 4.0	70-79 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z zaawansowanymi metodami programowania
NA OCENĘ 4.5	80-89 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z zaawansowanymi metodami programowania
NA OCENĘ 5.0	90-100 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z zaawansowanymi metodami programowania
EFEKT KSZTAŁCENIA 5	
NA OCENĘ 2.0	Poniżej 50 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z reprezentacjami i metodami generowania obiektów kombinatorycznych oraz wybranymi metodami generacji liczb losowych.
NA OCENĘ 3.0	50-59 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z reprezentacjami i metodami generowania obiektów kombinatorycznych oraz wybranymi metodami generacji liczb losowych
NA OCENĘ 3.5	60-69 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z reprezentacjami i metodami generowania obiektów kombinatorycznych oraz wybranymi metodami generacji liczb losowych
NA OCENĘ 4.0	70-79 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z reprezentacjami i metodami generowania obiektów kombinatorycznych oraz wybranymi metodami generacji liczb losowych
NA OCENĘ 4.5	80-89 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z reprezentacjami i metodami generowania obiektów kombinatorycznych oraz wybranymi metodami generacji liczb losowych

NA OCENĘ 5.0	90-100 % punktów uzyskanych za posiadaną wiedzę dotyczącą zagadnień związanych z reprezentacjami i metodami generowania obiektów kombinatorycznych oraz wybranymi metodami generacji liczb losowych
EFEKT KSZTAŁCENIA 6	
NA OCENĘ 2.0	Poniżej 50 % punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru klasycznych metod programowania.
NA OCENĘ 3.0	50-59 % punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru klasycznych metod programowania.
NA OCENĘ 3.5	60-69 % punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru klasycznych metod programowania.
NA OCENĘ 4.0	70-79 % punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru klasycznych metod programowania.
NA OCENĘ 4.5	80-89 % punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru klasycznych metod programowania.
NA OCENĘ 5.0	90-100 % punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru klasycznych metod programowania.
EFEKT KSZTAŁCENIA 7	
NA OCENĘ 2.0	Poniżej 50 % punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru zaawansowanych metod programowania.
NA OCENĘ 3.0	50-59 % punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru zaawansowanych metod programowania.
NA OCENĘ 3.5	60-69 % punktów uzyskanych za punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru zaawansowanych metod programowania.
NA OCENĘ 4.0	70-79 % punktów uzyskanych za punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru zaawansowanych metod programowania.
NA OCENĘ 4.5	80-89 % punktów uzyskanych za punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru zaawansowanych metod programowania.
NA OCENĘ 5.0	90-100 % punktów uzyskanych za punktów uzyskanych za napisanie i uruchomienie programów w języku C/C++ dla poznanego zbioru zaawansowanych metod programowania.
EFEKT KSZTAŁCENIA 8	
NA OCENĘ 2.0	Poniżej 50 % punktów uzyskanych za napisanie i uruchomienie programów generujących obiekty kombinatoryczne i liczby losowe.
NA OCENĘ 3.0	50-59 % punktów uzyskanych za napisanie i uruchomienie programów generujących obiekty kombinatoryczne i liczby losowe.

NA OCENĘ 3.5	60-69 % punktów uzyskanych za napisanie i uruchomienie programów generujących obiekty kombinatoryczne i liczby losowe.
NA OCENĘ 4.0	70-79 % punktów uzyskanych za napisanie i uruchomienie programów generujących obiekty kombinatoryczne i liczby losowe.
NA OCENĘ 4.5	80-89 % punktów uzyskanych za napisanie i uruchomienie programów generujących obiekty kombinatoryczne i liczby losowe.
NA OCENĘ 5.0	90-100 % punktów uzyskanych za napisanie i uruchomienie programów generujących obiekty kombinatoryczne i liczby losowe.
EFEKT KSZTAŁCENIA 9	
NA OCENĘ 2.0	Bierność lub niewywiązywanie się z przyjętych obowiązków lub destrukcyjny wpływ na pracę zespołu. Ocena poniżej 50 %.
NA OCENĘ 3.0	Umiarkowana aktywność lub niewywiązanie się z części przyjętych obowiązków lub brak kreatywności lub brak współpracy w zespole. Ocena 50-59 %.
NA OCENĘ 3.5	Zadowolająca aktywność, wywiązywanie się z przyjętych obowiązków, przejawy kreatywności, poprawna współpraca w zespole w roli wykonawcy (łącznie). Ocena 60-69 %.
NA OCENĘ 4.0	Dobra aktywność, wywiązywanie się z przyjętych obowiązków, kreatywność, efektywna współpraca w zespole (łącznie). Ocena 70-79 %.
NA OCENĘ 4.5	Dobra aktywność, wywiązywanie się z przyjętych obowiązków, kreatywność, efektywna współpraca w zespole, transfer wiedzy do pozostałych członków zespołu (łącznie). Ocena 80-89 %.
NA OCENĘ 5.0	Wyróżniająca się aktywność, wywiązywanie się z przyjętych obowiązków, kreatywność, efektywna współpraca w zespole, transfer wiedzy do pozostałych członków zespołu, wykazanie się umiejętnościami kierowniczymi (łącznie). Ocena 90-100 %.

10 MACIERZ REALIZACJI PRZEDMIOTU

EFEKT KSZTAŁCENIA	ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁOWYCH EFEKTÓW ZDEFINIOWANYCH DLA PROGRAMU	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	NARZĘDZIA DYDAKTYCZNE	SPOSOBY OCENY
EK1	K_W01 K_W03 K_U01 K_U04 K_K03 K_K04	Cel 1	W1 W15 K1	N1 N2	F1 P1

EFEKT KSZTAŁCENIA	ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁOWYCH EFEKTÓW ZDEFINIOWANYCH DLA PROGRAMU	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	NARZĘDZIA DYDAKTYCZNE	SPOSOBY OCENY
EK2	K_W03 K_W06 K_U04 K_U08 K_K01 K_K02	Cel 2 Cel 3 Cel 4	W2 K1 K2 K3 K4 K5 K6 K7 K8 K9 K11	N1 N2	F1 F2 F3 P1 P2 P3
EK3	K_W01 K_W03 K_U01 K_U04 K_K02 K_K03	Cel 2 Cel 3 Cel 4	W3 W4 W5 W6 W7 W8 W9 W10 W11 K1 K2 K3 K4 K5 K6 K7 K11	N1 N2 N3	F1 F2 F3 P1 P2 P3
EK4	K_W15 K_W23 K_U08 K_U10 K_K04 K_K06	Cel 3 Cel 4 Cel 5	W12 W13 W14 K2 K8 K9 K11	N1 N2 N3	F1 F2 F3 P1 P2 P3
EK5	K_W17 K_W26 K_U19 K_U21 K_K03 K_K06	Cel 2 Cel 3 Cel 4 Cel 5	W4 W6 W7 W10 W12 K1 K2 K11	N1 N2	F1 F2 P1 P2 P3
EK6	K_W03 K_W07 K_U04 K_U21	Cel 2 Cel 3 Cel 4	W2 W3 W4 W5 W8 W9 W10 W11 K3 K4 K5 K6 K7 K11	N2	F2 P2
EK7	K_W23 K_W26 K_U04 K_U08 K_K03 K_K06	Cel 2 Cel 3 Cel 4 Cel 5	W12 W13 W14 K8 K9 K11	N2 N3	F2 F3 P2
EK8	K_W17 K_W19 K_U10 K_U19 K_K03 K_K06	Cel 3 Cel 4	W2 W4 W6 W7 K1 K2 K11	N2 N3	F2 P2
EK9	K_W26 K_U02 K_U05 K_K02 K_K06	Cel 5	W12 W14 K9 K11	N3	F3 P2

11 WYKAZ LITERATURY

LITERATURA PODSTAWOWA

- [1] | Cormen T.H., Leiserson C.E., Rivest R.L., C. Stein — *Wprowadzenie do algorytmów*, Warszawa, 2007, WN-T
- [2] | Dasgupta S., Papadimitriou C., Vazirani U. — *Algorytmy*, Warszawa, 2010, PWN
- [3] | Lipski W. — *Kombinatoryka dla programistów*, Warszawa, 2004, WN-T

- [4] de Berg M., van Kreveld M., Overmars M., Schwarzkopf O. — *Geometria obliczeniowa. Algorytmy i zastosowania*, Warszawa, 2006, WN-T
- [5] Kernighan B.W., Pike R. — *Lekcja programowania*, Warszawa, 2002, WN-T

LITERATURA UZUPEŁNIAJĄCA

- [1] Sedgewick R. — *Algorytmy w C++. Sortowanie i wyszukiwanie*, Warszawa, 1999, Wydawnictwo RM
- [2] Sedgewick R. — *Algorytmy w C++. Algorytmy grafowe*, Warszawa, 2003, Wydawnictwo RM
- [3] Rauber T., Ruenger G. — *Parallel programming for multicore and cluster systems*, 3rd ed., 2019, Springer

12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

OSOBA ODPOWIEDZIALNA ZA KARTĘ

dr hab. inż. Prof PK Zbigniew Kokosiński (kontakt: zk@pk.edu.pl)

OSOBY PROWADZĄCE PRZEDMIOT

- 1 dr hab. inż. Zbigniew Kokosiński (kontakt: zk@pk.edu.pl)
- 2 mgr inż. Grzegorz Nowakowski (kontakt: gnowakowski@pk.edu.pl)

13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejscowość, data)

(odpowiedzialny za przedmiot)

(dziekan)

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....
.....