

# POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

## KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2012/2013

Wydział Inżynierii Elektrycznej i Komputerowej

Kierunek studiów: Elektrotechnika

Profil: Ogólnoakademicki

Forma studiów: niestacjonarne

Kod kierunku: Elek

Stopień studiów: II

Specjalności: Informatyczne systemy automatyki

### 1 INFORMACJE O PRZEDMIOCIE

NAZWA PRZEDMIOTU	Algorytmy i struktury danych
NAZWA PRZEDMIOTU W JĘZYKU ANGIELSKIM	Algorithms and Data Structures
KOD PRZEDMIOTU	WIEiK ELEKTROTECH oIIN PW15 12/13
KATEGORIA PRZEDMIOTU	Przedmioty specjalnościowe
LICZBA PUNKTÓW ECTS	4.00
SEMESTRY	3

### 2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

SEMESTR	WYKŁADY	ĆWICZENIA	LABORATORIA	LABORATORIA KOMPUTERO- WE	PROJEKTY	
3	15	0	0	10	5	0

### 3 CELE PRZEDMIOTU

**Cel 1** Poznanie podstawowych struktur danych: tablice, listy, kolejki, stosy, drzewa, grafy.

**Cel 2** Poznanie algorytmów wykorzystujących podstawowe struktury danych. Algorytmy sortowania i wyszukiwania; algorytmy grafowe (przeszukiwanie włąb, przeszukiwanie wszerek, drzewa rozpinające, ścieżki, skojarzenia, kolorowanie grafów).

**Cel 3** Poznanie zaawansowanych struktury danych: kolejki priorytetowe, kopce, drzewa zrównoważone.

**Cel 4** Poznanie algorytmów wykorzystujących zaawansowane struktury danych.

**Cel 5** Wykonanie indywidualnego projektu programistycznego w języku C++ : wykorzystanie wybranych struktur danych przy implementacji wybranego algorytmu.

**Cel 6** Poznanie wybranych problemów kombinatorycznych znajdujących zastosowanie w projektowaniu układów elektronicznych.

## 4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

1 Zaliczony przedmiot z pierwszego semestru: Programowanie w C++.

## 5 EFEKTY KSZTAŁCENIA

**EK1 Wiedza** Znajomość podstawowych i wybranych zaawansowanych struktur danych i ich zastosowań (tablice, listy, stos, kolejki priorytetowe, grafy, drzewa, kopce).

**EK2 Wiedza** Znajomość poznanego zbioru algorytmów wykorzystujących podstawowe i zaawansowane struktury danych.

**EK3 Umiejętności** Zrozumienie znaczenia doboru właściwych struktur danych dla skonstruowania efektywnego algorytmu.

**EK4 Umiejętności** Napisanie i uruchomienie programów w języku C/C++ dla poznanych algorytmów i struktur danych.

**EK5 Umiejętności** Znajomość wybranych problemów kombinatorycznych spotykanych w projektowaniu systemów elektronicznych.

## 6 TREŚCI PROGRAMOWE

WYKŁADY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
<b>W1</b>	Podstawowe struktury danych. Tablica. Lista i jej warianty: kolejka, stos, kolejka podwójna. Zbiór. Graf. Drzewo. Kopiec. Kolejka priorytetowa. Struktury. Unie. Abstrakcyjne typy danych.	2
<b>W2</b>	Wybrane algorytmy tablicowe i listowe. Sortowanie i wyszukiwanie.	2
<b>W3</b>	Podstawowe algorytmy grafowe: przeszukiwanie włąb, przeszukiwanie wszere, drzewa rozpinające, ścieżki, skojarzenia w grafach i kolorowanie grafów. Problemy podziałowe w grafach.	4
<b>W4</b>	Zaawansowane algorytmy grafowe: przepływy w sieciach, algorytm DMKM, problem komiwojażera. Problemy permutacyjne w grafach.	4
<b>W5</b>	Generacja obiektów kombinatorycznych. Podstawowe typy obiektów: kombinacje, permutacje, nieporządki, podziały, drzewa itp. Algorytmy generacyjne. Zastosowania : wyszukiwanie wyczerpujące, testowanie algorytmów kombinatorycznych.	2

WYKŁADY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
<b>W6</b>	Problemy algorytmiczne w projektowaniu układów elektronicznych.	1

LABORATORIA KOMPUTEROWE		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
<b>K1</b>	Algorytmy sortowania i wyszukiwania.	4
<b>K2</b>	Algorytmy grafowe.	4
<b>K3</b>	Generatory obiektów kombinatorycznych.	2

PROJEKTY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
<b>P1</b>	Projekt indywidualny: implementacja algorytmu wykorzystującego wybrane struktury danych.	5

## 7 NARZĘDZIA DYDAKTYCZNE

**N1** Wykłady

**N2** Ćwiczenia laboratoryjne

**N3** Ćwiczenia projektowe

## 8 OBCIĄŻENIE PRACĄ STUDENTA

FORMA AKTYWNOŚCI	ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI
<b>Godziny kontaktowe z nauczycielem akademickim, w tym:</b>	
Godziny wynikające z planu studiów	0
Konsultacje przedmiotowe	10
Egzaminy i zaliczenia w sesji	10
<b>Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym:</b>	
Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	25
Opracowanie wyników	0
Przygotowanie raportu, projektu, prezentacji, dyskusji	15
programowanie, testowanie programów	30
<b>SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA</b>	<b>90</b>
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	4.00

## 9 SPOSOBY OCENY

### OCENA FORMUJĄCA

F1 Projekt indywidualny

F2 Sprawozdanie z ćwiczenia laboratoryjnego

### OCENA PODSUMOWUJĄCA

P1 Średnia ważona ocen formujących

P2 Zaliczenie pisemne

### WARUNKI ZALICZENIA PRZEDMIOTU

W1 Obecności na wykładach i laboratoriach komputerowych

### OCENA AKTYWNOŚCI BEZ UDZIAŁU NAUCZYCIELA

B1 Projekt indywidualny

### KRYTERIA OCENY

EFEKT KSZTAŁCENIA 1

NA OCENĘ 2.0	Brak znajomości przynajmniej 4 podstawowych struktur danych i ich zastosowań (lista jedno- i dwukierunkowa, stos, tablica, graf).
NA OCENĘ 3.0	Znajomość 4 podstawowych struktur danych i ich zastosowań (lista jedno- i dwukierunkowa, stos, tablica, graf).
NA OCENĘ 3.5	Znajomość 5 podstawowych struktur danych i ich zastosowań (lista jedno- i dwukierunkowa, stos, kolejka priorytetowa, tablica, graf).
NA OCENĘ 4.0	Znajomość 6 podstawowych struktur danych i ich zastosowań (lista jedno- i dwukierunkowa, stos, kolejka priorytetowa, tablica, graf, drzewo binarne).
NA OCENĘ 4.5	Znajomość 7 podstawowych struktur danych i ich zastosowań (lista jedno- i dwukierunkowa, stos, kolejka priorytetowa, tablica, graf, drzewo binarne, kopiec).
NA OCENĘ 5.0	Znajomość 8 podstawowych struktur danych i ich zastosowań (lista jedno- i dwukierunkowa, stos, kolejka priorytetowa, tablica, graf, drzewo binarne, drzewo t-arne, kopiec).
EFEKT KSZTAŁCENIA 2	
NA OCENĘ 2.0	Brak znajomości teoretycznej przynajmniej 2 spośród 5 wybranych algorytmów (Mergesort/QuickSort, BFS/DFS, HeapSort, algorytm Prim'a, algorytm DMKM).
NA OCENĘ 3.0	Znajomość 2 spośród 5 wybranych algorytmów (Mergesort/QuickSort, BFS/DFS, HeapSort, algorytm Prim'a, algorytm DMKM).
NA OCENĘ 3.5	Znajomość 3 spośród 5 wybranych algorytmów (Mergesort/QuickSort, BFS/DFS, HeapSort, algorytm Prim'a, algorytm DMKM).
NA OCENĘ 4.0	Znajomość 4 spośród 5 wybranych algorytmów (Mergesort/QuickSort, BFS/DFS, HeapSort, algorytm Prim'a, algorytm DMKM).
NA OCENĘ 4.5	Znajomość 5 wybranych algorytmów (Mergesort/QuickSort, BFS/DFS, HeapSort, algorytm Prim'a, algorytm DMKM).
NA OCENĘ 5.0	Znajomość 5 wybranych algorytmów (Mergesort/QuickSort, BFS/DFS, HeapSort, algorytm Prim'a, algorytm DMKM) oraz konstrukcji generatorów obiektów kombinatorycznych.
EFEKT KSZTAŁCENIA 3	
NA OCENĘ 2.0	Brak zrozumienia znaczenia doboru właściwych struktur danych dla skonstruowania efektywnego algorytmu.
NA OCENĘ 3.0	Umiejętność podania jednego przykładu optymalnego doboru właściwej struktury danych dla skonstruowania efektywnego algorytmu.
NA OCENĘ 3.5	Umiejętność podania dwóch przykładów optymalnego doboru właściwej struktury danych dla skonstruowania efektywnego algorytmu, bez umiejętności uogólnienia przykładów.

NA OCENĘ 4.0	Umiejętność podania co najmniej trzech przykładów optymalnego doboru właściwej struktury danych dla skonstruowania efektywnego algorytmu, bez umiejętności uogólnienia przykładów.
NA OCENĘ 4.5	Umiejętność podania co najmniej trzech przykładów optymalnego doboru właściwej struktury danych dla skonstruowania efektywnego algorytmu, oraz umiejętność uogólnienia przykładów.
NA OCENĘ 5.0	Umiejętność podania co najmniej trzech przykładów optymalnego doboru właściwej struktury danych dla skonstruowania efektywnego algorytmu, oraz umiejętność uogólnienia przykładów. Podanie przykładu z własnej praktyki programistycznej.
EFEKT KSZTAŁCENIA 4	
NA OCENĘ 2.0	Brak umiejętności napisania i uruchomienia zadanego programu w języku C/C++ .
NA OCENĘ 3.0	Napisanie i uruchomienie zadanego programu w języku C/C++ z dużą ilością błędów.
NA OCENĘ 3.5	Napisanie i uruchomienie zadanego programu w języku C/C++ z niewielką ilością błędów i z niepełną funkcjonalnością.
NA OCENĘ 4.0	Napisanie i uruchomienie zadanego programu w języku C/C++ bez błędów i z wystarczającą funkcjonalnością. Poprawna obsługa formatów we/wy.
NA OCENĘ 4.5	Napisanie i uruchomienie zadanego programu w języku C/C++ bez błędów i z pełną funkcjonalnością. Poprawna obsługa formatów we/wy.
NA OCENĘ 5.0	Napisanie i uruchomienie zadanego programu w języku C/C++ bez błędów i z pełną funkcjonalnością. Poprawna obsługa formatów we/wy. Interfejs graficzny użytkownika, generacja raportów.
EFEKT KSZTAŁCENIA 5	
NA OCENĘ 2.0	Brak znajomości problemów kombinatorycznych spotykanych w projektowaniu systemów elektronicznych.
NA OCENĘ 3.0	Znajomość 1 problemu kombinatorycznego spotykanego w projektowaniu systemów elektronicznych.
NA OCENĘ 3.5	Znajomość 2 problemów kombinatorycznych spotykanych w projektowaniu systemów elektronicznych.
NA OCENĘ 4.0	Znajomość 3 problemów kombinatorycznych spotykanych w projektowaniu systemów elektronicznych.
NA OCENĘ 4.5	Znajomość 4 problemów kombinatorycznych spotykanych w projektowaniu systemów elektronicznych.
NA OCENĘ 5.0	Znajomość 5 problemów kombinatorycznych spotykanych w projektowaniu systemów elektronicznych.

## 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_W02, K_W03	Cel 1 Cel 3 Cel 5	W3 K1 K2 K3 P1	N1 N2 N3	F1 F2 P1
EK2	K_W02, K_W03	Cel 2 Cel 4 Cel 5	W2 W4 W5 K1 K2 K3 P1	N1 N2 N3	F1 F2 P1
EK3	K_U01, K_U11	Cel 1 Cel 2 Cel 3 Cel 4 Cel 5	W1 W2 W3 W4 W5 K1 K2 K3 P1	N1 N2 N3	F1 F2 P1
EK4	K_U11	Cel 5	W2 W4 K1 K2 K3 P1	N1 N2 N3	F1 F2 P1
EK5	K_U03, K_U11	Cel 6	W6	N1	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 ] Kernighan B.W., Pike R. — *Lekcja programowania*, Warszawa, 2002, WN-T
- [3 ] Lipski W. — *Kombinatoryka dla programistów*, Warszawa, 2004, 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

## 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 inż. Zbigniew Kokosiński (kontakt: Zbigniew.Kokosinski@pk.edu.pl)



## 13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

---

(miejsowość, data)

(odpowiedzialny za przedmiot)

(dziekan)

**PRZYJMUJĘ DO REALIZACJI** (data i podpisy osób prowadzących przedmiot)

.....