

KARTA PRZEDMIOTU

1. Informacje ogólne

| | |
|---|---|
| Nazwa przedmiotu i kod (wg planu studiów): | Algorytmy i struktury danych, C3 |
| Nazwa przedmiotu (j. ang.): | Algorithms and data structures |
| Kierunek studiów: | Informatyka |
| Specjalność/specjalizacja: | Sieciowe systemy informatyczne, Technologie internetowe i bazy danych, Informatyka praktyczna |
| Poziom kształcenia: | studia I stopnia |
| Profil kształcenia: | praktyczny (P) |
| Forma studiów: | studia stacjonarne/ studia niestacjonarne |
| Obszar kształcenia: | nauki techniczne (wg wykazu) |
| Dziedzina: | nauki techniczne (wg wykazu) |
| Dyscyplina nauki: | informatyka |
| Koordinator przedmiotu: | Prof. Barbara Dębska |

2. Ogólna charakterystyka przedmiotu

| | |
|---|---|
| Przynależność do modułu: | kształcenia kierunkowego |
| Status przedmiotu: | obowiązkowy |
| Język wykładowy: | polski |
| Rok studiów, semestr: | I, 2 |
| Forma i wymiar zajęć według planu studiów: | stacjonarne - wykład 30 h, ćw. audytoryjne 30 h niestacjonarne - wykład 30 h, ćw. audytoryjne 15 h |
| Interesariusze i instytucje partnerskie (nieobowiązkowe) | |
| Wymagania wstępne: | Podstawy budowy algorytmów, programowanie w języku Pascal, obliczanie pochodnych i granic funkcji, |

3. Bilans punktów ECTS

| Całkowita liczba punktów ECTS (wg planu studiów; 1 punkt =25-30 godzin pracy studenta, w tym praca na zajęciach i poza zajęciami): | 5 (A + B) | stacjonarne | Niestacjonarne |
|---|---|---|--|
| | | A. Liczba godzin wymagających bezpośredniego udziału nauczyciela z podziałem na typy zajęć oraz całkowita liczba punktów ECTS osiągniętych na tych zajęciach | Wykład Ćwiczenia audytoryjne W sumie: ECTS |
| B. Poszczególne typy zadań do samokształcenia studenta (niewymagających bezpośredniego udziału nauczyciela) wraz z planowaną średnią liczbą godzin na każde i sumaryczną liczbą ECTS | przygotowanie ogólne praca w bibliotece praca na platformie e-learningowej w sumie: ECTS | 10 30 30 70 2 | 10 30 45 85 2 |
| C. Liczba godzin praktycznych/laboratoryjnych w ramach przedmiotu oraz związana z tym liczba punktów ECTS | | | |

4. Opis przedmiotu

| |
|--|
| <p>Cel przedmiotu: Celem przedmiotu jest nauka praktycznego projektowania i analizy algorytmów. Dokonany jest także przegląd struktur danych dynamicznych oraz algorytmów sortowania.</p> |
| <p>Metody dydaktyczne: wykład, ćwiczenia przedmiotowe, ćwiczenia z użyciem komputera.</p> |
| <p>Treści kształcenia: Wykłady: Podstawowe zasady analizy algorytmów. Złożoność obliczeniowa. Pesymistyczna i oczekiwana złożoność czasowa, oczekiwana wrażliwość czasowa. Definicja rzędu wielkości funkcji. Najczęściej spotykane złożoności algorytmów. Obliczanie złożoności czasowej algorytmów metodą równań rekurencyjnych. Poprawność semantyczna programu. Model obliczeniowy ze swobodnym dostępem do pamięci. Maszyna RAM. Opis komend (zestaw poleceń) maszyny RAM. Programowanie w języku maszyny RAM. Pseudokod. Programowanie w pseudokodzie. Złożoność pamięciowa i czasowa programów napisanych w języku maszyny RAM oraz w pseudokodzie. Podstawowe struktury danych. Statyczne typy danych: proste i złożone. Dynamiczne struktury informacyjne: lista, kolejka, stos. Podstawowe operacje wykonywane na listach, kolejkach i stosach. Zapis infiksowy i postfiksowy. Zbiory i ich implementacje listowe. Struktury grafowe. Przedstawianie grafów za pomocą list i macierzy sąsiedztwa. Drzewa. Drzewa binarne. Metody przechodzenia drzew binarnych. Drzewa regularne i pełne. Użycie struktur wskaźnikowych do kodowania drzew. Rekurencyjne wykonywanie algorytmów. Algorytmy przeszukiwania drzew. Kopce. Kolejki priorytetowe. Algorytmy przechodzenia drzew. Strategia „wszerz” i strategia „w głąb”. Sortowanie. Algorytmy oparte na porównaniach. Algorytmy asymptotycznie optymalne. Metoda sortowania „dziel i zwyciężaj”. Sortowanie przez scalanie. Sortowanie w czasie liniowym: sortowanie przez zliczanie i sortowanie „kubelkowe”. Czas działania i złożoność obliczeniowa algorytmów sortowania. Realizacja struktur dynamicznych za pomocą języka PASCAL. Dynamiczny dostęp do pamięci. Dostęp listowy. Deklaracja list. Procedury wykonywania działań na listach. Deklarowanie w języku PASCAL struktur typu kolejka, stos i kopiec, drzewo binarne. Algorytmy wykonywania operacji na tych strukturach: czytanie, wstawianie, porządkowanie, poszukiwanie i wyświetlanie zawartości.</p> |

Ćwiczenia audytoryjne:

1. Szacowanie złożoności obliczeniowej algorytmów – metodą algorytmiczną.
2. Szacowanie złożoności obliczeniowej algorytmów metodą porównywania rzędów wielkości funkcji.
3. Szacowanie złożoności obliczeniowej algorytmów metodą sprowadzania równań rekurencyjnych do sumy.
4. Maszyna RAM – programowanie w języku maszyny RAM, obliczanie złożoności czasowych i obliczeniowych programów zapisanych w języku maszyny RAM.
5. Dynamiczne struktury danych.
6. Implementacja wybranych struktur dynamicznych w języku Turbo Pascal.
7. Algorytmy sortowania.

5. Efekty kształcenia i sposoby weryfikacji**Efekty kształcenia:**

K_U01,K_U02,K_U03,K_U08,K_U10,K_U12,K_U13,K_U27,K_U30,K_U31

| Efekt przedmiotu <i>(kod przedmiotu + kod efektu kształcenia)</i> | Student, który zaliczył przedmiot (spełnił minimum wymagań) | Efekt kierunkowy |
|--|---|---|
| C3_K_W06 C3_K_W08 | Wiedza: 1. Ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie algorytmów i ich złożoności obliczeniowej 2. Zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu analizy złożoności obliczeniowej algorytmów | K_W06 K_W08 |
| C3_K_U01 C3_K_U02 C3_K_U03 C3_K_U08 C3_K_U10 C3_K_U13 C3_K_U30 | Umiejętności 1. Potrafi wykorzystać nabytą wiedzę matematyczną do opisu procesów, tworzenia modeli, zapisu algorytmów 2. Wykorzystuje wiedzę matematyczną do optymalizacji rozwiązań zarówno sprzętowych jak i programowych; potrafi wykorzystać do formułowania i rozwiązywania zadań informatycznych metody analityczne i eksperymentalne 3. Zna metody samokształcenia i umie korzystać z dydaktycznych portali internetowych 4. Potrafi wykorzystać do formułowania i rozwiązywania zadań informatycznych proste metody analityczne i eksperymentalne, w tym proste eksperymenty obliczeniowe 5. Ma umiejętność formułowania algorytmów i ich programowania z użyciem przynajmniej jednego z typowych narzędzi. 6. Potrafi ocenić złożoność obliczeniową algorytmów i problemów 7. Potrafi ocenić, na podstawowym poziomie, przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do typowych zadań informatycznych. | K_U01 K_U02 K_U03 K_U08 K_U10 K_U13 K_U30 |

| | | | | |
|---|--|---|--|--|
| C3_K_K08 | Kompetencje społeczne 1. Rozumie potrzebę wsparcia analizę modeli matematycznych stosownymi narzędziami informatycznymi. | | | K_K08 |
| Sposoby weryfikacji efektów kształcenia: | | | | |
| Lp. | Efekt przedmiotu | Sposób weryfikacji | Ocena formująca – przykładowe sposoby jej wystawienia poniżej | Ocena końcowa przykładowe sposoby jej wystawienia poniżej |
| 1 | C3_K_W06 C3_K_W08 | Aktywność studenta na zajęciach. Sprawdzian pisemny ze znajomości metod szacowania złożoności algorytmów. | Ocena z odpowiedzi ustnych, ocena z częściowych prac pisemnych | Średnia z ocen formujących, Ocena podsumowująca: ocena z egzaminu |
| | C3_K_U01 C3_K_U02 C3_K_U03 C3_K_U10 C3_K_U13 | Aktywność studenta na zajęciach. Sprawdzian pisemny ze znajomości metod szacowania złożoności algorytmów, z umiejętności programowania w języku RAM, ze znajomości struktur dynamicznych oraz algorytmów sortowania. Aktywność na portalu dydaktycznym e-Student. | Ocena z odpowiedzi ustnych, ocena z częściowych prac pisemnych | Średnia z ocen Ocena podsumowująca: ocena z egzaminu z egzaminu |
| | C3_K_K08 | Aktywność studenta na zajęciach. | Ocena z odpowiedzi ustnych | Średnia z ocen formujących, Ocena podsumowująca: ocena z egzaminu |
| Kryteria oceny: | | | | |
| w zakresie wiedzy | | | | Efekt kształcenia |
| Na ocenę 3,0 | Student uzyskał minimum 50% punktów ze sprawdzianów pisemnych | | | C3_K_W06 C3_K_W08 |
| Na ocenę 5,0 | Student uzyskał ponad 90% punktów ze sprawdzianów pisemnych | | | C3_K_W06 C3_K_W08 |
| w zakresie umiejętności | | | | |
| Na ocenę 3,0 | Student uzyskał minimum 50% punktów ze sprawdzianów pisemnych | | | |
| Na ocenę 5,0 | Student uzyskał ponad 90% punktów ze sprawdzianów pisemnych | | | |
| w zakresie kompetencji społecznych | | | | |
| Na ocenę 3,0 | Student uzyskał minimum 50% punktów ze sprawdzianów pisemnych | | | |
| Na ocenę 5,0 | Student uzyskał ponad 90% punktów ze sprawdzianów pisemnych | | | |

Kryteria oceny formującej:

aktywność za zajęciach: 20%
ocena ze sprawdzianów pisemnych: 80%

Kryteria oceny podsumowującej:

ocena z egzaminu: 100%

Zalecana literatura:

1. Wirth N., Algorytmy + struktury danych = programy, Wydawnictwa Naukowo-Techniczne , Warszawa, 2001
2. Banachowski L., Diks K., Rytter W., Algorytmy i struktury danych, Wydawnictwa Naukowo-Techniczne, Warszawa, 1999
3. Aho A.V., Algorytmy i struktury danych, Helion, Gliwice, 2003
4. Wróblewski P., Algorytmy, struktury danych i techniki programowania, Helion, Gliwice, 1997
5. Ochodek B., Algorytmy i struktury danych, Państwowa Wyższa Szkoła Zawodowa , Piła, 2003

Informacje dodatkowe:

| Dodatkowe obowiązki prowadzącego wraz z szacowaną całkowitą liczbą godzin: |
|---|
| Przygotowanie do wykładów i ćwiczeń audytoryjnych – 30 godzin |
| Konsultacje – 15 godzin |
| Poprawa kolokwium– 40 godzin |
| Przygotowanie ćwiczeń e-learningowych - 15 godzin |
| Przygotowanie i poprawa egzaminu – 5 godzin |
| W sumie: 105 godzin |

