

KARTA PRZEDMIOTU DLA NABORU 2020/2021

INFORMACJE OGÓLNE

1. Nazwa przedmiotu kształcenia

ALGORYTMY I STRUKTURY DANYCH

2. Nazwa jednostki prowadzącej moduł Zakład Informatyki

3. Grupa treści kształcenia -

4. Typ przedmiotu obowiązkowy

5. Poziom studiów pierwszego stopnia

6. Liczba punktów ECTS 5

7. Poziom przedmiotu średniozaawansowany

8. Rok studiów, semestr II rok, semestr III

9. Liczba godzin w semestrze laboratoria 30h, wykłady 30 h

10. Język wykładowy: polski

11. Wykładowca (wykładowcy)

Marcin Klimek, dr inż.

INFORMACJE SZCZEGÓŁOWE

12. Wymagania wstępne

1. Znajomość elementów matematyki dyskretniej tj. teoria grafów itp.
2. Wiedza z podstaw analizy matematycznej.
3. Umiejętność programowania w zakresie podstawowym.

13. Cele przedmiotu

C1. Zapoznanie z podstawowymi strukturami danych i możliwościami ich wykorzystania w programach komputerowych.

C2. Zdobyć wiedzy dotyczącej technik algorytmicznych i możliwości ich zastosowania w różnych problemach optymalizacyjnych.

C3. Zaznajomienie z podstawowymi algorytmami dla ważnych zagadnień informatycznych tj. sortowanie, wyszukiwanie danych itp.

C4. Zdobyć umiejętności zapisu procedur obliczeniowych w postaci pseudokodu, schematów blokowych.

C5. Zrozumienie zasad oceny przydatności algorytmu w konkretnym zastosowaniu m.in. ze względu na złożoność obliczeniową i pamięciową.

14. Efekty uczenia się w zakresie wiedzy, umiejętności i kompetencji społecznych

Student, który zaliczył przedmiot potrafi:	odniesienie do kierunkowych efektów uczenia się
--	---

WIEDZA

EU01 zna i rozumie podstawowe techniki analizy algorytmów	I1P_W07 I1P_W13
EU02 ma wiedzę dotyczącą standardowych struktur danych	
EU03 ma wiedzę dotyczącą podstawowych algorytmów	

UMIEJĘTNOŚCI

EU04 umie implementować wydajne algorytmy, stosować odpowiednie techniki algorytmiczne dla praktycznych problemów obliczeniowych	I1P_U12 I1P_U14
EU05 potrafi oceniać złożoność obliczeniową zastosowanych procedur obliczeniowych identyfikować trudne zagadnienia obliczeniowe	
EU06 umie wykorzystywać podstawowe struktury danych	

KOMPETENCJE SPOŁECZNE

EU07 rozumie potrzebę ciągłego dokształcania się	I1P_K01 I1P_K04
EU08 potrafi krytycznie ocenić przedstawione rozwiązania algorytmiczne, uzasadnić swoją opinię podczas dyskusji w grupie	

15. Treści programowe

Forma zajęć – wykłady

W1 Wprowadzenie do algorytmiki. Definicje algorytmu. Cechy algorytmu. Sposoby zapisu algorytmów. Liczby pseudolosowe.

W2 Złożoność obliczeniowa. Problemy NP-trudne i NP-zupełne.

W3 Metody tworzenia algorytmów: dziel i zwyciężaj, metoda zachłanna, programowanie dynamiczne,

	rekurencja, heurystyki itd.
W4	Przykładowe problemy optymalizacyjne i wybrane procedury ich rozwiązywania (problem plecakowy, wydawania reszty, komiwojażera, wieża Hanoi itp.).
W5	Algorytmy sortowania (szybkie, przez wstawianie, wybieranie, bąbelkowe, pozycyjne, kopcowe)
W6	Analiza złożoności wybranych algorytmów (np. sortowania).
W7	Typy danych: tablice, stosy, kolejki, listy jedno- i dwukierunkowe. Podstawowe operacje na
W8	Struktury danych w C#. Efektywne wyszukiwanie (binarne), haszowanie. Słowniki.
W9	Grafy. Cykl Eulera, Hamiltona. Algorytm Dijkstry.
W10	Drzewa. BST, AVL, B drzewa. Kopiec. Podstawowe operacje na drzewach.
Forma zajęć – laboratoria	
L1	Implementacja w C# prostych algorytmów np. znajdowanie liczb pierwszych, szukanie minimum funkcji, operacje na macierzach itp.
L2	Kodowanie w C# podstawowych operacji na listach, tablicach tj. wyszukiwanie, dodawanie, usuwanie elementów.
L3	Konstruowanie algorytmów rekurencyjnych i iteracyjnych (wyszukiwanie binarne, obliczanie silni, ciąg Fibonacciego, wieża Hanoi).
L4	Budowa w C# algorytmów zachłannych i siłowych dla przykładowych problemów tj. problem wydawania reszty, plecakowy, komiwojażera itp.
L5	Budowa klasycznych algorytmów sortowania : bąbelkowy, przez wstawianie, wybieranie, szybkie sortowanie. Analiza złożoności obliczeniowej.
L6	Budowa dynamicznych struktur danych: stosów, kolejek, list i podstawowych operacji dla tych struktur. Korzystanie z kolekcji w C#.
L7	Implementacja w C# algorytmów przeszukiwania drzew, grafów, algorytmu Dijkstry.
16. Narzędzia/metody dydaktyczne	
1.	Podczas laboratoriów praca indywidualna studentów: pisanie programów komputerowych, rozwiązywanie problemów algorytmicznych, analiza kodu programów komputerowych pod kątem wydajności, optymalności.
2.	Laboratoria w części odbywają się przy tablicy: studenci rozwiązują zagadnienia teoretyczne tj. sortowanie liczb, obliczanie złożoności obliczeniowej.
3.	Wykład prowadzony z wykorzystaniem projektora multimedialnego. Wykłady przygotowane w formie prezentacji PowerPoint.
4.	Podczas wykładów prezentowane są także wybrane programy komputerowe w C# tj. generowanie liczb pseudolosowych (Lotto), wyszukiwanie binarne, przeglądanie drzew, sortowanie szybkie, przez wstawianie, implementacje stosu, kolejki, listy dwukierunkowej, przykłady obsługi kolekcji Stack, Queue, Dictionary, SortedList, algorytm Dijkstry, generowanie wszystkich permutacji n liczb itp.
17. Sposoby oceny (F – formująca; P – podsumowująca)	
F1.	Dwa kolokwia przy komputerze: implementacja programów stosujących różne struktury danych i techniki algorytmiczne.
F2.	Kolokwium pisemne z złożoności obliczeniowej algorytmów.
F3.	Ocena realizacji projektu zaliczeniowego do zrealizowania w domu (implementacja wybranego algorytmu rozwiązującego problem optymalizacyjny).
F4.	Kolokwium pisemne z algorytmów sortowania
F5.	Ocena ciągła pracy indywidualnej (realizacji programów komputerowych) podczas laboratoriów.
P1.	Egzamin w formie pisemnej sprawdzający wiedzę z wykładów i laboratoriów
18. Obciążenia pracą studenta	
forma aktywności	średnia liczba godzin na zrealizowanie aktywności
Godziny kontaktowe z nauczycielem*	60
Nakład pracy studenta	
Przygotowanie się do zajęć	20
Przygotowanie projektu zaliczeniowego	30
Przygotowanie do egzaminu	20
SUMA	130
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	5
19. Literatura podstawowa i uzupełniająca	
Literatura podstawowa:	
1)	Wirth N., Algorytmy + struktury danych = programy. WNT 2004
2)	Cormen Thomas H. Algorytmy bez tajemnic, Gliwice : Helion, 2013

- 3) Wróblewski P., Algorytmy, struktury danych i techniki programowania. Helion 2010.
- 4) Lafore R., Java. Algorytmy i struktury danych. Helion 2004.
- 5) Knuth D. Sztuka programowania. Tom 1. Algorytmy podstawowe, Tom 2. Algorytmy suminumeryczne. Warszawa: WNT 2003.

Literatura uzupełniająca:

- 1) Banachowski L., Diks K., Rytter W., Algorytmy i struktury danych. WNT 1996.
- 2) Cormen T. H., Leiserson C. E., Rivest R. L., Stein C., Wprowadzenie do algorytmów. WNT Warszawa 2004.
- 3) Stephens R., Algorytmy i Struktury Danych. Helion Gliwice 2000.
- 4) Dasgupta S., Papadimitriou Ch., Vazirani U.; Algorytmy, Warszawa : Wydawnictwo Naukowe PWN, 2010.

20. Formy oceny - szczegóły

Warunki uzyskania zaliczenia przedmiotu: zajęcia kończą się zaliczeniem z oceną.

Sposób weryfikacji efektów uczenia się:

Ocena stopnia osiągniętych przez studenta efektów uczenia się następuje wg poniższych kryteriów:

- 5.0 – zakładany efekt uczenia się został osiągnięty bez zastrzeżeń
- 4.5 – zakładany efekt uczenia się został osiągnięty z pojedynczymi brakami/błędami
- 4.0 – zakładany efekt uczenia się został osiągnięty z nielicznymi brakami/błędami
- 3.5 – zakładany efekt uczenia się został osiągnięty z wieloma brakami/błędami
- 3.0 – zakładany efekt kształcenia został osiągnięty z licznymi i istotnymi brakami/błędami (minimalnie wymagany poziom osiągnięcia efektu)
- 2.0 – zakładany efekt uczenia się nie został osiągnięty

21. Inne przydatne informacje o przedmiocie

1. Bezpośrednich informacji o problematyce zajęć i treściach programowych udziela prowadzący w trakcie zajęć i podczas konsultacji
2. Zajęcia odbywać się będą w PSW w Białej Podlaskiej
3. Zajęcia odbywać się będą zgodnie z aktualnym planem zajęć
4. Konsultacje odbywać się będą zgodnie z obowiązującym terminarzem
5. Materiały z wykładów i laboratoriów udostępnione są dla studentów na stronie psw.na11.pl

* Zajęcia prowadzone z bezpośrednim udziałem nauczyciela akademickiego lub innej osoby prowadzącej zajęcia oraz konsultacje