



UNIwersYTET
IM. ADAMA MICKIEWICZA
W POZNANIU

Algorytmy i struktury danych Sylabus zajęć

Informacje podstawowe

Kierunek studiów Bioinformatyka	Cykl dydaktyczny 2023/24
Specjalność -	Kod zajęć 01BINS.12N.00327.23
Jednostka organizacyjna Wydział Biologii	Języki wykładowe Polski
Poziom studiów Studia pierwszego stopnia	Obligatoryjność Obowiązkowy
Forma studiów Studia stacjonarne	Blok zajęciowy Przedmioty nieprzypisane
Profil studiów Profil ogólnoakademicki	
Koordynator zajęć	Stanisław Gawiejnowicz
Prowadzący zajęcia	Stanisław Gawiejnowicz, Michał Hanćkowiak

Okres Semestr 2	Forma zajęć / liczba godzin / forma zaliczenia <ul style="list-style-type: none">Wykład: 30, Zaliczenie z ocenąLaboratorium: 30, Zaliczenie z oceną	Liczba punktów ECTS 5
---------------------------	---	---------------------------------

Cele kształcenia dla zajęć

Kod	Cel
C1	Zapoznanie studentów z podstawowymi statycznymi i dynamicznymi strukturami danych.
C2	Zapoznanie studentów z zasadami analizy złożoności czasowej oraz pamięciowej algorytmów.
C3	Przekazanie studentom podstawowej wiedzy na temat problemów NP-trudnych oraz metodyki ich rozwiązywania.
C4	Przekazanie studentom podstawowej wiedzy na temat implementacji algorytmów oraz zasad postępowania się strukturami danych.

Wymagania wstępne

Podstawowa znajomość języka Python, pozwalająca na pisanie średnio zaawansowanych programów.

Efekty uczenia się dla zajęć

Kod	Efekty uczenia się dla zajęć w zakresie	Efekty uczenia się dla kierunku	Metody weryfikacji osiągnięcia efektów uczenia się dla zajęć
Wiedzy - Student/ka:			
W1	podstawowe statyczne oraz dynamiczne struktury danych	BIN_K1_W01, BIN_K1_W02	Test
W2	podstawowe metody analizy złożoności czasowej i pamięciowej algorytmów	BIN_K1_W01, BIN_K1_W02	Test
W3	pojęcie problemu NP-trudnego oraz jego konsekwencje	BIN_K1_W01, BIN_K1_W02	Test
Umiejętności - Student/ka:			
U1	zaimplementować wybrane algorytmy w języku Python	BIN_K1_U02, BIN_K1_U03	Kolokwium pisemne, Projekt
U2	napisać pseudokod wybranych algorytmów	BIN_K1_U01, BIN_K1_U02, BIN_K1_U03	Kolokwium pisemne, Projekt

Treści programowe dla zajęć

Lp.	Treści programowe dla zajęć	Efekty uczenia się dla zajęć	Formy zajęć
1.	Podstawowe pojęcia i definicje: problem algorytmiczny; algorytm: budowa, własności, sposoby zapisu; operacje elementarne; typ danych; pseudokod i jego składowe: nagłówek, operacje wejścia/wyjścia, zdanie warunkowe, pętle sterowane licznikiem, pętle warunkowe; program, zasady testowania; maszyna Turinga; problemy nierozstrzygalne	W1, U2	Wykład, Laboratorium
2.	Algorytmy elementarne 1: zmienna, stała; algorytmy wykorzystujące jedynie zmienne i stałe: obliczanie pól i objętości; tablice jednowymiarowe; algorytmy wykorzystujące tablice jednowymiarowe: sumy i iloczyny składowych wektora, obliczanie potęgi, binarny algorytm obliczania potęgi, schemat Hornera obliczania wartości wielomianu w punkcie, wyszukiwanie w nieuprządkowanej tablicy, wyszukiwanie binarne; programowanie dynamiczne, liczby Fibonacciego	W1, U1, U2	Wykład, Laboratorium
3.	Algorytmy elementarne 2: tablice dwuwymiarowe; algorytmy wykorzystujące tablice dwuwymiarowe: operacje na macierzach; algorytmy dokonywania obliczeń z zadaną dokładnością; algorytm Herona obliczania pierwiastka kwadratowego, obliczanie rozwinięcia dziesiętnego liczby e	W1, U1, U2	Wykład, Laboratorium

4.	Funkcje i procedury: pojęcia makra i podprogramu; procedura: budowa, własności; funkcja: budowa, własności; parametry i argumenty funkcji oraz procedur, metody przekazywania argumentów: przez wartość, przez zmienną; zmienne lokalne i globalne	W1, U1, U2	Wykład, Laboratorium
5.	Rekurencja i algorytmy rekurencyjne: rekurencja: warunek początkowy, równanie rekurencyjne, rząd rekurencji; przykłady definicji rekurencyjnych: silnia, największy wspólny dzielnik, współczynnik Newtona, liczby Fibonacci'ego; wady rekurencji: funkcja Ackermanna; graficzne zastosowania rekurencji: fraktale	W1, U1	Wykład, Laboratorium
6.	Algorytmy sortowania 1: notacja asymptotyczna; problem sortowania: definicja, sortowanie a relacja częściowego porządku; sortowanie przez wybieranie; sortowanie przez wstawianie; sortowanie bąbelkowe; przykład analizy algorytmu sortowania	W1, W2, U1	Wykład, Laboratorium
7.	Algorytmy sortowania 2: technika 'dziel i zwyciężaj'; sortowanie przez scalanie; sortowanie szybkie; dolne oszacowanie złożoności problemu sortowania; sortowanie przez zliczanie	W1, W2, U1	Wykład, Laboratorium
8.	Złożoność algorytmów: złożoność obliczeniowa i złożoność pamięciowa algorytmu; optymistyczna, średnia i pesymistyczna złożoność obliczeniowa algorytmu; algorytmy wielomianowe, pseudowielomianowe oraz wykładnicze; zasady analizy złożoności algorytmów; złożoność wybranych algorytmów i problemów	W1, W2, U1	Wykład, Laboratorium
9.	Stosy, kolejki i listy: statyczne i dynamiczne struktury danych; wskaźniki i atrybuty; stos: budowa, podstawowe operacje; kolejka: budowa, podstawowe operacje; listy pojedynczo wiązane: budowa, podstawowe operacje; listy podwójnie wiązane: budowa, podstawowe operacje	W1, W2, U1	Wykład, Laboratorium
10.	Struktury drzewiaste: drzewa, drzewa ukorzenione; drzewa uporządkowane; drzewo binarne: budowa, własności; drzewo BST: budowa, własności; metody przechodzenia drzewa BST: preorder, inorder, postorder; operacje na drzewie BST: wyszukiwanie, budowanie, wstawianie, usuwanie; implementacja słownika; drzewa RB jako ulepszenie BST (podstawowe informacje)	W1, W2, U1	Wykład, Laboratorium
11.	Kopce binarne, kolejki priorytetowe: kopiec binarny: definicja, budowa, kopiec typu max, kopiec typu min; operacje na kopcu binarnym; kolejka priorytetowa: definicja, budowa; operacje na kolejce priorytetowej; kolejka priorytetowa bazująca na kopcu binarnym vs na drzewie RB	W1, W2, U1	Wykład, Laboratorium
12.	Algorytmy grafowe 1: grafy skierowane i nieskierowane; metody reprezentacji grafu: macierz sąsiedztwa, macierz incydencji, listy sąsiedztwa, listy krawędzi; listowe i drzewiaste struktury dla zbiorów rozłącznych; operacje na zbiorach rozłącznych; przeszukiwanie grafu: w głąb, w szerz; sortowanie topologiczne grafu	W1, W2, U1, U2	Wykład, Laboratorium

13.	Algorytmy grafowe 2: problem cyklu Eulera: definicja, twierdzenia Eulera, algorytm Fleury'ego; problem cyklu Hamiltona: definicja, złożoność; problem minimalnego drzewa rozpinającego: definicja, algorytm Prima, algorytm Kruskala; problem najkrótszych ścieżek z jednym źródłem, algorytmy Dijkstry i Bellmana-Forda	W1, W2, W3, U2	Wykład, Laboratorium
14.	Teoretyczne modele obliczeń: jednotaśmowa deterministyczna maszyna Turinga DTM: budowa, funkcja przejścia, przykład programu; funkcje nieobliczalne w sensie Turinga; klasa P problemów rozwiązywalnych w wielomianowym czasie przez DTM; wielotaśmowa deterministyczna maszyna Turinga; maszyna RAM: budowa, lista rozkazów, przykład programu; związki między modelami obliczeń; niedeterministyczna maszyna Turinga NDTM: budowa, definicja; klasa NP problemów rozwiązywalnych w wielomianowym czasie przez NDTM; związki między klasami P i NP, hipoteza P=?NP	W2	Wykład
15.	Problemy NP-trudne: problemy NP-zupełne, metodyka dowodzenia NP-zupełności; problem spełnialności wyrażeń boolowskich SAT; główne problemy NP-zupełne: problem 3-SAT, problem 3-wymiarowego skojarzenia, problem pokrycia wierzchołkowego, problem podziału zbioru, problem cyklu Hamiltona, problem klik, problem wędrującego komiwojażera; NP-zupełność a NP-trudność; konsekwencje NP-zupełności i NP-trudności	W2, W3	Wykład
16.	Zaawansowane metody projektowania algorytmów: algorytmy zachłanne i algorytmy dynamiczne; przykłady algorytmów dynamicznych: problem najkrótszego wspólnego ciągu, problem optymalnego nawiasowania przy mnożeniu macierzy	W1, W2	Wykład

Informacje dodatkowe

Forma zajęć	Metody i formy prowadzenia zajęć
Wykład	Wykład z prezentacją multimedialną wybranych zagadnień, Demonstracje dźwiękowe i/lub video
Laboratorium	Rozwiązywanie zadań (np.: obliczeniowych, artystycznych, praktycznych), Metoda ćwiczeniowa, Metoda laboratoryjna

Forma zajęć	Warunki zaliczenia zajęć
Wykład	Zaliczenie ewaluacyjnego testu wielokrotnego wyboru z materiału wykładu. Skala ocen: poniżej 50% punktów= 2.0, od 50% do 60%=3.0, od 60% do 70% = 3.5, od 70% do 80% = 4.0, od 80% do 90% = 4.5, powyżej 90% = 5.0.
Laboratorium	Zaliczenie dwu pisemnych kolokwii z materiału laboratorium oraz kilku projektów programowych wymagających napisania implementacji w języku Python. Skala ocen: poniżej 50% punktów= 2.0, od 50% do 60%=3.0, od 60% do 70% = 3.5, od 70% do 80% = 4.0, od 80% do 90% = 4.5, powyżej 90% = 5.0.

Literatura

Obowiązkowa

1. A.V. Aho, J.E. Hopcroft, J.D. Ullman: Projektowanie i analiza algorytmów komputerowych, WNT, 1983.
2. T.H. Cormen, C.E. Leiserson, R.L. Rivest: Wprowadzenie do algorytmów, WNT, 2012.
3. N. Wirth: Algorytmy+struktury danych=programy, WNT, 2000.

Dodatkowa

1. J.E. Hopcroft, J.D. Ullman: Wprowadzenie do teorii automatów, języków i obliczeń, WN PWN, 1994.
2. D.E. Knuth, Sztuka programowania, tomy 1-3, WNT, 2002.
3. W. Lipski: Kombinatoryka dla programistów, WNT, 1989.
4. C.H. Papadimitriou: Złożoność obliczeniowa, WNT, 2002.
5. V. Vazirani: Algorytmy aproksymacyjne, WNT, 2005.

Nakład pracy studenta i punkty ECTS

Rodzaje zajęć studenta	Średnia liczba godzin* przeznaczonych na zrealizowane rodzaje zajęć
Wykład	30
Laboratorium	30
Przygotowanie projektu	30
Przygotowanie do egzaminu	15
Przygotowanie do zaliczenia	15
Czytanie wskazanej literatury	15
Łączny nakład pracy studenta	Liczba godzin 135
Liczba punktów ECTS	ECTS 5

* godzina (lekcyjna) oznacza 45 minut

Efekty uczenia się dla kierunku

Kod	Treść
BIN_K1_U01	Absolwent/ka potrafi stosować metody matematyczne i statystyczne do opisu i symulacji procesów biologicznych
BIN_K1_U02	Absolwent/ka potrafi dobierać i stosować metody informatyczne do opisywania zjawisk i procesów biologicznych oraz analizy danych biologicznych
BIN_K1_U03	Absolwent/ka potrafi formułować i rozwiązywać problemy biologiczne z zastosowaniem narzędzi bioinformatycznych
BIN_K1_W01	Absolwent/ka zna i rozumie metody matematyczne i statystyczne w zakresie niezbędnym do opisu i analizy danych biologicznych
BIN_K1_W02	Absolwent/ka zna i rozumie metody, techniki i narzędzia informatyczne stosowane w rozwiązywaniu problemów biologicznych