



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Pracownia algorytmiczna

Przedmiot

Kierunek studiów

Bioinformatyka

Studia w zakresie (specjalność)

-

Poziom studiów

pierwszego stopnia

Forma studiów

stacjonarne

Rok/semestr

3/5

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

30

Laboratoria

30

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

4

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Marcin Radom

Odpowiedzialny za przedmiot/wykładowca:

Wymagania wstępne

Student rozpoczynający ten moduł powinien posiadać podstawową wiedzę nt. algorytmów i struktur danych oraz klas złożoności obliczeniowej. Dodatkowo powinien posiadać umiejętność konstrukcji algorytmów, ich prawidłowej implementacji oraz analizowania jakości zwracanych przez nie wyników. Wymagane jest aby student przystępujący do przedmiotu posiadał dobre umiejętności programowania obiektowego w językach C++/Java.

Cel przedmiotu

1. Rozszerzenie wiedzy na temat algorytmów bioinformatycznych i zastosowanie ich do rozwiązywania problemów optymalizacyjnych.
2. Rozwinięcie u studentów umiejętności analizy złożoności badanych problemów i konstrukcji efektywnych rozwiązań.
3. Zapoznanie z publikacjami naukowymi dostarczającymi aktualnego stanu wiedzy w badanych problemach.



4. Przekazanie studentom wiedzy z zakresu języka Java oraz projektowania aplikacji za pomocą schematów UML.
5. Stworzenie wspólnej aplikacji z możliwością porównania oraz wizualizacji wyników zaprojektowanych algorytmów.
6. Rozwinięcie u studentów umiejętności programowania obiektowego do celów implementacji algorytmów

Przedmiotowe efekty uczenia się

Wiedza

W wyniku przeprowadzonych zajęć student:

1. Zna zagadnienia z zakresu algorytmów i struktur danych omawianych na laboratoriach, potrzebnych do implementacji algorytmów, zna również podstawy teorii złożoności obliczeniowej w celu doboru odpowiedniej klasy algorytmu.
2. Zna podstawowe problemy optymalizacyjne, dla których poszukuje odpowiednich rozwiązań.
3. Zna zasady programowania obiektowego w językach Java / C#.

Umiejętności

W wyniku przeprowadzonych zajęć student:

1. Potrafi pozyskiwać informacje z literatury oraz innych właściwie dobranych źródeł (serwisy internetowe poświęcone językom programowania i zagadnieniom pochodnym), także w języku angielskim.
2. Samodzielnie zdobywa wiedzę i podnosi swoje kwalifikacje, niezbędne z uwagi na często zmieniające się style i podejścia do programowania w środowisku Visual Studio.
3. Projektuje i tworzy oprogramowanie komputerowe zgodnie z zadaną specyfikacją, używając właściwych metod, technik i narzędzi.

Kompetencje społeczne

Zaliczenie przedmiotu oznacza, że student:

1. Rozumie potrzebę uczenia się przez całe życie i podnoszenia swoich kompetencji z uwagi na ciągłe rozwijanie języków programowania o wciąż nowe i przydatne funkcjonalności.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca

- a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 - odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach



b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę i „obronę” zrealizowanych przez studenta ćwiczeń laboratoryjnych
- ocenę wiedzy i umiejętności związanych z realizacją zadań laboratoryjnych poprzez dwa sprawdziany pisemne w semestrze

Ocena podsumowująca

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę wiedzy i umiejętności wykazanych na kolokwium pisemnym w formie testu wielokrotnego wyboru oraz nielicznych pytań otwartych. Kolokwium składa się z 15 pytań o łącznej wartości 30 punktów rozdzielonych w zależności od stopnia trudności pytania. Ocenę pozytywną studenci otrzymują po zdobyciu minimum połowy punktów.
- omówienie wyników kolokwium

b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- w trakcie realizacji zajęć laboratoryjnych studenci mają do napisania kolokwium dotyczące zagadnień algorytmicznych realizowanych na wykładzie oraz laboratorium. Dodatkowo muszą napisać program zaliczeniowy zgodnie ze specyfikacją przedstawioną na zajęciach.

Aktywność podczas zajęć premiuwana jest dodatkowymi punktami, w szczególności za:

- efektywność zastosowania zdobytej wiedzy podczas pisania zadanego programu w sposób wykraczający poza minimum określone w specyfikacji

Treści programowe

Program wykładu obejmuje zagadnienia dotyczące teorii algorytmów oraz podstawowych problemów optymalizacyjnych, których implementacja odbywać się będzie w języku C#. Na wykładach studenci zapoznają się z:

- składnią, obiektami oraz stylami programowania w języku C# oraz środowiskiem Visual Studio,
- klasami złożoności obliczeniowej, przykładami algorytmów dokładnych i heurystycznych,
- podstawowymi problemami optymalizacyjnymi (np. problem pakowania, szeregowania, znajdowania ścieżki w grafie),
- algorytmami rozwiązującymi złożone kombinatoryczne problemy spotykane w bioinformatyce.

Ćwiczenia laboratoryjne prowadzone są w formie piętnastu dwugodzinnych zajęć. Pierwsze zajęcia przeznaczone są na zapoznanie studentów z zasadami użytkowania laboratorium i zaliczenia ćwiczeń. Program kolejnych zajęć laboratoryjnych obejmuje następujące zagadnienia:

- implementacja przykładów przedstawionych na wykładach,



- pisanie samodzielne programów w ramach rozwiązywania wybranych problemów kombinatorycznych,
- implementacja większego projektu w dwuosobowej grupie,
- ćwiczenia dotyczące dobrych praktyk programistycznych potrzebnych do efektywnej implementacji zaproponowanych rozwiązań.

Metody dydaktyczne

1. Wykład: prezentacja multimedialna oraz dodatkowe przykłady podawane na tablicy w miarę potrzeb.
2. Ćwiczenia laboratoryjne: rozwiązywanie zadań, ćwiczenia praktyczne, praca w zespole.

Literatura

Podstawowa

Andrew Troelsen, "Pro C# 5.0 and the .NET 4.5 Framework", 2012.

Wprowadzenie do algorytmów, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Wydawnictwa Naukowo - Techniczne, 2004

Uzupełniająca

Algorytmy i struktury danych, L. Banachowski, K. Diks, W. Rytter, Wydawnictwa Naukowo - Techniczne, 2006.

Złożoność obliczeniowa algorytmów i problemów szeregowania zadań, Jacek Błażewicz, Wydawnictwo Politechniki Poznańskiej, 1979.

Złożoność obliczeniowa problemów kombinatorycznych, Jacek Błażewicz, Wydawnictwa Naukowo-Techniczne, 1988-

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	100	4,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	50	2,0
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu) ¹	50	2,0

¹ niepotrzebne skreślić lub dopisać inne czynności