



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Biologiczne aplikacje baz danych [S2Bioinf2>BABD]

Przedmiot

Kierunek studiów
Bioinformatyka

Rok/Semestr
1/1

Studia w zakresie (specjalność)
–

Profil studiów
ogólnoakademicki

Poziom studiów
drugiego stopnia

Język oferowanego przedmiotu
polski

Forma studiów
stacjonarne

Wymagalność
obligatoryjny

Liczba godzin

Wykład
30

Laboratorium
30

Inne
0

Ćwiczenia
0

Projekty/seminaria
0

Liczba punktów ECTS

4,00

Koordynatorzy

dr inż. Anna Leśniewska
anna.lesniewska@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z podstaw programowania, systemów operacyjnych oraz algebry. Powinien posiadać umiejętność budowania zapytań w języku SQL, aby móc swobodnie komunikować się z bazą danych. Powinien posiadać umiejętność pozyskiwania informacji ze wskazanych źródeł. Powinien również rozumieć konieczność poszerzania swoich kompetencji oraz mieć gotowość do podjęcia współpracy w ramach zespołu. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, ciekawość poznawcza, kreatywność, kultura osobista oraz szacunek dla innych ludzi.

Cel przedmiotu

Przedstawienie studentom podstawowej wiedzy z technologii systemów baz danych niezbędnej do poprawnego zaprojektowania i implementowania systemów baz danych oraz ich aplikacji. Rozwijanie u studentów umiejętności rozwiązywania problemów pojawiających się w pracy z systemami baz danych. Rozwinięcie u studentów umiejętności budowania zaawansowanych zapytań w języku SQL oraz podprogramów w PL/SQL do uzyskania potrzebnej wiedzy biologicznej z danych.

Przedmiotowe efekty uczenia się

Wiedza:

- Ma podstawową wiedzę o cyklu życia systemów baz danych
- Zna narzędzia i techniki z zakresu przetwarzania i analizy danych w bazach danych wykorzystywane do rozwiązywania zadań bioinformatycznych

Umiejętności:

- Potrafi korzystać z materiałów w postaci literatury, materiałów dydaktycznych oraz dokumentacji technicznej systemów baz danych zarówno w języku polskim jak i angielskim
- Stosuje poznane narzędzia i techniki zaawansowanych analiz danych w bazie danych do rozwiązywania problemów biologicznych
- Wykonuje zadania badawcze pod kierunkiem prowadzącego, planując i wykorzystując metody analityczne na danych w bazie danych
- Formuluje i testuje hipotezy związane z problemami bioinformatycznymi oraz weryfikuje je podczas analizy
- Wykorzystuje zdobyte umiejętności do tworzenia oprogramowania np. w postaci generowania formularzy i raportów, oraz oprogramowania do zarządzania i manipulowania danymi w bazie danych
- Potrafi programować w języku SQL i PL/SQL

Kompetencje społeczne:

- Rozumie, że wiedza z zakresu baz danych zmienia się bardzo dynamicznie i istnieje potrzeba systematycznego aktualizowania swojej wiedzy i umiejętności w tej dziedzinie
- Jest gotów do odpowiedzialnego pełnienia ról zawodowych, z uwzględnieniem podtrzymywania etosu zawodu, oraz do przestrzegania zasad etyki zawodowej i działania na rzecz przestrzegania tych zasad

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wiedza nabyta w ramach wykładu jest weryfikowana przez kolokwium zaliczeniowe realizowane na 15 wykładzie. Kolokwium składa się z 15-20 pytań (testowych i otwartych), różnie punktowanych w zależności od stopnia trudności. Próg zaliczeniowy: 50% punktów. Zagadnienia zaliczeniowe, na podstawie których opracowane są pytania zostaną wcześniej udostępnione studentom.

Umiejętności zdobyte w ramach zajęć laboratoryjnych weryfikowane są na podstawie zestawienia ocen wystawianych w trakcie semestru w postaci średniej ważonej (1/2 realizacja ćwiczeń na zajęciach) oraz przygotowanie i prezentację przez studenta aplikacji bazodanowej (1/2 ocena z projektu).

Treści programowe

W ramach wykładu studenci poznają następujące zagadnienia:

- Wprowadzenie do baz danych i specyfika baz danych biologicznych
- Modelowanie pojęciowe
- Transformacja schematu pojęciowego do schematu implementacyjnego
- Relacyjny model danych
- Programowanie w języku PL/SQL i wykorzystanie w analizie danych bioinformatycznych
- Metody uwierzytelniania użytkowników
- Zasady przyznawania uprawnień i sposoby autoryzacji operacji w bazie danych
- Struktury indeksowe w bazach danych
- Interfejsy programowe do baz danych oraz wizualizacja danych biologicznych składowanych w bazie danych
- Problemy badawcze w pracy z biologicznymi bazami danych
- Bazy nurtu NoSQL a biologiczne bazy danych

Ćwiczenia laboratoryjne są praktycznym odzwierciedleniem treści prezentowanych na wykładzie i obejmują następujące zagadnienia:

- SQL do zaawansowanych analiz bioinformatycznych (grupowanie GROUP BY i HAVING, kostka i półkostka danych ROLLUP i CUBE, transformacja danych wierszowych do układu kolumnowego PIVOT i UNPIVOT)
- Programowanie w języku PL/SQL - wybrane elementy (koncepcja języka, zmienne i stałe, typy zmiennych, nadawanie wartości zmiennym, instrukcje warunkowe, pętle, sterowanie przebiegiem programu, kursory, wyjątki, podprogramy składowane - procedury, funkcje i pakiety)
- Metody uwierzytelniania użytkowników
- Zasady przyznawania uprawnień i sposoby autoryzacji operacji w bazie danych

- Indeksy - charakterystyka, zastosowanie i pielęgnacja
- Wybrane technologie dostępu do relacyjnych baz danych (JDBC, R, Python) i wykorzystanie ich w pracy z biologicznymi bazami danych
- Projektowanie aplikacji bazodanowej w środowisku Oracle APEX
- Praca z dokumentową bazą danych na przykładzie MongoDB

Tematyka zajęć

Zajęcia przedstawiają kluczowe tematy związane z pracą i analizą danych biologicznych, składowanych w relacyjnych i nierelacyjnych bazach danych:

Zaawansowany SQL

- Grupowanie (GROUP BY) i filtracja (HAVING): instrukcje kluczowe do agregacji danych. GROUP BY grupuje wiersze z tych samych wartości, a HAVING filtruje wyniki po agregacji.
- Kostka i półkostka danych (ROLLUP i CUBE): Umożliwiają tworzenie złożonych raportów wielowymiarowych poprzez generowanie podsumowań na różnych poziomach hierarchii.
- Transformacja danych (PIVOT i UNPIVOT): PIVOT konwertuje dane z wierszy na kolumny, co ułatwia analizę. UNPIVOT działa odwrotnie, przekształcając kolumny na wiersze.

PL/SQL

- Rozszerzenie języka SQL o Podprogramy składowane: Tworzenie procedur, funkcji i pakietów pozwala na fragmentację kodu i jego ponowne wykorzystanie.

Uwierzytelnianie i Autoryzacja

- Metody uwierzytelniania oraz przyznawanie uprawnień: Ważne dla bezpieczeństwa danych.

Technologie dostępu do baz danych wykorzystujące różne środowiska programowe:

- JDBC: Umożliwia łączenie języka Java z bazami danych.
- R i Python: Używane szeroko w bioinformatyce do analizy danych.

Projektowanie aplikacji w Oracle APEX

- Oracle Application Express: Narzędzie do tworzenia prostych aplikacji webowych, opartych na współpracy z bazą danych.

Praca z MongoDB

- Dokumentowe bazy danych na przykładzie MongoDB jako przedstawiciela baz nurtu NoSQL, która przechowuje dane w formacie dokumentów JSON. Umożliwia elastyczne modelowanie danych i efektywne zarządzanie dużymi zbiorami danych.

Metody dydaktyczne

1. Wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, demonstracja.
2. Ćwiczenia laboratoryjne: rozwiązywanie zadań i implementowanie rozwiązań, ćwiczenia praktyczne, praca w zespole, pokaz multimedialny.

Literatura

Podstawowa:

1. Garcia-Molina H., Ullman J.D., Widom J., Implementacja systemów baz danych, WNT, 2003
2. Ullman J.D., Widom J., Podstawowy wykład z systemów baz danych, WNT, W-wa, 2000
3. Elmasri R., Navathe S., Wprowadzenie do systemów baz danych, Helion, (4th Edition), 2005
4. Wrembel R., Wieczerzycki W., Projektowanie aplikacji baz danych Oracle. NAKOM.
5. Feuerstein S., Pribyl B., Dawes C., Język Oracle PL/SQL. Leksykon podstawowy, Helion
6. Matloff N., The Art of R Programming, No Starch Press, 2011

Uzupełniająca:

1. Braun, W., & Murdoch, D. (2007). A First Course in Statistical Programming with R. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511803642
2. Quick John M., Analiza statystyczna w środowisku R dla początkujących, Helion

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	100	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	40	1,50