Kod: **02 53 5076 00** Liczba punktów ECTS: **2**

Nazwa przedmiotu	**Inteligentne przetwarzanie tekstu**
Nazwa w języku angielskim	
Język prowadzenia zajęć	polski
Poziom studiów	studia II stopnia
Profil studiów	A, ogólnoakademicki
Jednostka prowadząca	Instytut Informatyki Stosowanej
Kierownik i realizatorzy	**Grabowski Szymon, dr hab.** Bieniecki Wojciech, dr inż.

Formy zajęć i liczba godzin w semestrze

Wyk.	Ćw.	Lab.	Proj.	Sem.	Inne	Suma godzin w semestrze
15	0	15	0	0	0	30

Cele przedmiotu

po zmianie

Celem przedmiotu jest zapoznanie studentów z podstawowymi algorytmami przetwarzania tekstu, w trybie online i offline, oraz wybranymi ich zastosowaniami (m.in. aplikacje typu CAT, korekcja tekstu po OCR).

Efekty kształcenia

po zmianie

Student poznaje algorytmy specyficzne dla przetwarzania danych tekstowych.
Oczekiwanymi efektami kształcenia są:
1. znajomość sposobów przechowywania informacji tekstowej w bazach danych,
2. wyszukiwanie informacji tekstowej w sposób dokładny i przybliżonych w bazach danych i plikach,
3. znajomość algorytmów kompresji danych tekstowych, w tym algorytmów kompresji wspierających wyszukiwanie,
4. umiejętność tworzenia i wykorzystania miar podobieństwa do detekcji plagiatów i tłumaczenia (parsing, rozbiór zdania, analiza słów).

Metody weryfikacji efektów kształcenia

po zmianie

Efekty nr 2 i 3: samodzielne ćwiczenia laboratoryjne.
Efekty nr 1, 2 i 3: kolokwium wykładowe.
Efekt nr 4: prezentacja.

Wymagania wstępne

po zmianie

Algorytmy i struktury danych, Języki skryptowe, Metody i języki programowania.

Organizacja przedmiotu i treści kształcenia

po zmianie

WYKŁAD
1. Taksonomia problemów (wyszukiwanie on-line i off-line, wyszukiwanie dokładne i różne warianty wyszukiwania przybliżonego, szukanie wielu wzorców etc.).
2. Klasyczne algorytmy wyszukiwania dokładnego on-line: naiwny, KMP, BMH.
3. Miary wyszukiwania przybliżonego: odległość Hamminga, odległość Levenshteina, odległość indel, pasowanie (delta, gamma, alpha) i ich zastosowania, m.in. w bioinformatyce.
4. Miary podobieństwa sekwencji: najdłuższa wspólna podsekwencja (LCS), najdłuższy wspólny podciąg (LCSS).
5. Techniki algorytmiczne: programowanie dynamiczne, niedeterministyczny automat skończony i ich zastosowanie w wyszukiwaniu wzorców. Bit-parallelism.
6. Algorytmy wyszukiwania dokładnego on-line oparte na "równoległości bitowej": Shift-Or, BNDM, Average-Optimal Shift-Or.
7. Indeksowanie tekstu: drzewo sufiksowe (ST), tablica sufiksowa (SA). Wybrane algorytmy tworzenia SA. Transformata Burrowsa-Wheelera (BWT), jej związek z SA i zastosowania w kompresji danych.
8. Indeksowanie tekstu na poziomie słów: algorytm GLIMPSE.
9. Dyskusja i warianty schematu GLIMPSE (obsługa problemu krańców bloków, dodanie kompresji do reprezentacji list bloków), rozszerzenie jego funkcjonalności (obsługa zapytań przybliżonych).
10. Statystyczne modele języka. Dystrybucja słów w tekstach naturalnych. Prawo Zipfa i Heapsa.
11. Information Retrieval: zasady wyszukiwania dokumentów w bazie podobnych do danego zapytania (frazy). Pojęcia kompletności (recall) i dokładności (precision).
12. Popularne typy zapytań dla wyszukiwarek sieciowych.
13. PageRank.
14. Wykorzystanie technik statystycznego przetwarzania danych do analizy tekstu (wybrane zastosowania: wykrywanie plagiatów, ustalanie autorstwa dokumentu, klasyfikacja tematyczna dokumentów, zwalczanie email spamu i web spamu).

ĆWICZENIA LABORATORYJNE
1. Wykorzystanie wyrażeń regularnych.
2. Implementacja i testowanie algorytmów wyszukiwania przybliżonego w tekście.
3. Analiza logów systemowych.
4. Parsowanie dokumentów HTML (z wykorzystaniem gotowych bibliotek).
5. Empiryczna weryfikacja prawa Zipfa i prawa Heapsa.
6. Testowanie wybranych indeksów pełnotekstowych i opartych na słowach.

Formy zaliczenia - sprawdzenie osiągnięć efektów kształcenia

po zmianie

Ocena końcowa jest średnią ważoną oceny z egzaminu obejmującego materiał wykładu (40%) oraz oceny z zadań wykonywanych na ćwiczeniach laboratoryjnych (60%).

Literatura podstawowa

po zmianie

Cormen T., Leiserson Ch., Rivest R., Stein C.: Wprowadzenie do algorytmów. WNT, 2006.
Banachowski L., Diks K., Rytter W.: Algorytmy i struktury danych, WNT, Warszawa, 2003.

Literatura uzupełniająca

po zmianie

Witten I., Moffat A., Bell T.: Managing Gigabytes. Compressing and Indexing Documents and Images (2nd ed.). Morgan Kaufmann Publishing, 1999.
Mykowiecka A.: Inżynieria lingwistyczna. Komputerowe przetwarzanie tekstów w języku naturalnym. Wydawnictwo PJWSTK, 2007.
Baeza-Yates R., Ribeiro-Neto B.: Modern Information Retrieval. ACM Press, 1999.
Abiteboul S., Buneman P., Suciu D.: Dane w sieci WWW. Mikom, 2001.
Harris S., Ross J.: Algorytmy. Od podstaw. Helion, 2006.
Sayood K.: Kompresja danych. RM, 2002.

Przeciętne obciążenie studenta pracą własną - ze zdefiniowaniem form pracy własnej

Suma godzin wszystkich form zajęć	30
Udział w konsultacjach	5
Udział w pisemnych i/lub praktycznych formach weryfikacji	1
Przygotowywanie się do laboratorium	12
Przygotowywanie się do kolokwium wykładowego	6
Suma godzin:	54
Suma godzin powinna mieścić się w zakresie:	50..60

Uwagi

po zmianie

brak

Uwagi własne publikowane

Aktualizacja

2012-07-18

Code **02 53 5076 00** ECTS credits: **2**

Course name

Course name in Polish **Inteligentne przetwarzanie tekstu**

Language of instruction

Level of studies

Type of studies nie zdefiniowano

Unit running the programme Instytut Informatyki Stosowanej

Course coordinator and academic teachers

Grabowski Szymon, dr hab.

Bieniecki Wojciech, dr inż.

Form of classes and number of teaching hour per semester

Lec.	Tut.	Lab.	Proj.	Sem.	Other	Total number of teaching hour per semester
15	0	15	0	0	0	30

Goals

po zmianie

The aim of the course is to teach students basic text processing algorithms, working online and offline, and their selected applications (computer-aided translation applications, post-OCR text correction, etc.).

Learning outcomes

after changes

The student is expected to learn textual data processing analysis. His expected new knowledge and skills should comprise:
1. ways of storing information in text databases,
2. on-line and off-line text information search algorithms, for exact and approximate matching,
3. text compression algorithms, including those that support efficient query handling,
4. how to create and use similarity measures for plagiarism detection and translation support (sentence parsing, word stemming etc.).

Learning outcomes verification methods

after changes

Effects no 2 and 3: laboratory assigments.
Effects no 1, 2, and 3: lecture test.
Effect no 4: presentation.

Prerequisites

after changes

Algorithms and data structures, Script languages, Programming languages and methods.

Course organisation and content

after changes	LECTURE 1. The taxonomy of problems (on-line and off-line searching, exact and approximate searching, multiple pattern searching etc.). 2. Classic exact on-line search algorithms: naive, KMP, BMH. 3. Approximate search measures: Hamming distance, Levenshtein distance, indel distance, delta, gamma, alpha matching and their applications in bioinformatics and other fields. 4. The sequence similarity measures: longest common subsequence (LCS), longest common substring (LCSS). 5. Algorithmic techniques: dynamic programming, non-deterministic finite automata, and their applications to pattern searching. Bit-parallelism. 6. Bit-parallel algorithms for on-line exact searching: Shift-Or, BNDM, Average-Optimal Shift-Or. 7. Text indexing: suffix tree (ST), suffix array (SA). SA constructing algorithms. Burrows-Wheeler transform (BWT), its relation to SA and its data compression applications. 8. Word-based text indexing: GLIMPSE algorithm. 9. Discussion and variants of the GLIMPSE scheme (block boundary handling, adding compression to lists of indices), extending its functionalities (approximate query handling). 10. Statistical models of the language. Word distribution in NL texts. Zipf's and Heaps' laws. 11. Information Retrieval: how to retrieve documents relevant to a given query (phrase). Precision and Recall. 12. Common query types in web searches. 13. PageRank. 14. Statistical pattern recognition techniques in text analysis (selected applications: plagiarism detection, document authorship attribution, topic classification, fighting e-mail and web spam). LABORATORY 1. Regular expression applications. 2. Implementations and exprimental tests of selected approximate search algorithms. 3. System/web log analysis. 4. HMTL parsing (using available libraries). 5. Empirical verification of Zipf's and Heaps' laws. 6. Testing selected full-text and word-based indexes.
Form of assessment	
after changes	The final grade is the weighted average of the lecture (theory) exam grade (40%) oraz the laboratory assignments grade (60%).
Basic reference materials	
after changes	Cormen T., Leiserson Ch., Rivest R., Stein C.: Wprowadzenie do algorytmów. WNT, 2006. Banachowski L., Diks K., Rytter W.: Algorytmy i struktury danych, WNT, Warszawa, 2003.
Other reference materials	

after changes

Witten I., Moffat A., Bell T.: Managing Gigabytes. Compressing and Indexing Documents and Images (2nd ed.). Morgan Kaufmann Publishing, 1999.
Mykowiecka A.: Inżynieria lingwistyczna. Komputerowe przetwarzanie tekstów w języku naturalnym. Wydawnictwo PJWSTK, 2007.
Baeza-Yates R., Ribeiro-Neto B.: Modern Information Retrieval. ACM Press, 1999.
Abiteboul S., Buneman P., Suciu D.: Dane w sieci WWW. Mikom, 2001.
Harris S., Ross J.: Algorytmy. Od podstaw. Helion, 2006.
Sayood K.: Kompresja danych. RM, 2002.

Average student work-load outside classroom

Total hours of different forms of classes	30
Participation in consultations	5
Participation in written and/or practical forms of assesment	1
Preparation to laboratories	12
Preparation to the lecture test	6
Total hours:	54
Total hours should be in the range:	50..60

Published comments

Aktualizacja 2012-07-18