

KARTA KURSU

Nazwa	Teoretyczne Podstawy Informatyki
Nazwa w j. ang.	Theoretical Foundations of Computer Science

Koordynator	prof. dr hab. Jacek Migdałek	Zespół dydaktyczny
Punktacja ECTS*	st. stacjonarne: 6 st. niestacjonarne: 6	prof. dr hab. Jacek Migdałek dr hab. Tomasz Dobrowolski, prof. UP

Opis kursu (cele kształcenia)

Celem realizacji kursu jest zapoznanie studentów kierunku Informatyka z podstawami teoretycznymi informatyki: teorią informacji, zasadami kodowania informacji, gramatykami języków formalnych, problemami translacji języków programowania oraz elementami algorytmiki. Przygotowanie studentów do samodzielnego uczenia się i uzupełniania wiedzy w zakresie podstaw informatyki. Kurs prowadzony jest w języku polskim.

Warunki wstępne

Wiedza	Podstawowe wiadomości z matematyki i informatyki na poziomie szkoły średniej.
Umiejętności	Uruchamianie komputera, obsługa systemu operacyjnego Windows lub Linux
Kursy	Wstępne kursy nie są wymagane.

Efekty uczenia się

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Wiedza	W01: zna systemy liczbowe, arytmetykę binarną, podstawy asemantycznej teorii informacji i aspekty jej kodowania.	K_W01
	W02: scharakteryzuje i sklasyfikuje języki i gramatyki formalne, rozpoznaje problemy akceptacji i translacji języków formalnych i naturalnych, scharakteryzuje modele teoretyczne maszyn cyfrowych oraz ich praktyczne realizacje, w tym architekturę komputerów.	K_W01
	W03: zna pojęcie algorytmu, jego strukturę, cechy, sposoby zapisu, metody badania poprawności algorytmów, optymalizację złożoności czasowej i pamięciowej algorytmów, wskaże problemy obliczalności i rozstrzygalności algorytmów	K_W01

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Umiejętności	U01: Wyznacza entropię bezpamięciowego źródła informacji, potrafi skonstruować rozszerzenie źródła bezpamięciowego, posługuje się twierdzeniami o entropii źródeł rozszerzonych. Na podstawie macierzy prawdopodobieństw wyznacza graf dla źródła Markowa i vice-versa. Wyznacza entropie źródła Markowa w ustalonych stanach, wyznacza entropię źródła Markowa. Rozpoznaje kody nieosobliwe, jednoznacznie dekodowalne oraz dekodowalne bez opóźnienia. Dla danego kodu oraz źródła wyznacza średnią długość ciągu kodowego, sprawność oraz rozwlekłość kodu. Konstruuje zwarte kody binarne oraz r-narne metodą Huffmana. Stosuje metodę kodowania Shannona-Fano. Wyznacza redundancję kodu. Dla kanałów informacyjnych wyznacza prawdopodobieństwa sygnałów wyjściowych na podstawie prawdopodobieństw sygnałów wejściowych. Wyznacza entropię „a priori”, „a posteriori” oraz miarę nieoznaczoności kanału. Wyznacza odległość Hamminga ciągów kodowych. Przekształca zadany kod na kod z kontrolą parzystości. Konstruuje kod Hamminga i wykorzystuje go do korekty błędów.	K_U02 K_U16 K_U17
	U02: Zamienia liczby dziesiętne na binarne, oktalne oraz heksadecymalne i vice versa. Wykonuje działania arytmetyczne na liczbach binarnych (dodawanie, odejmowanie, mnożenie, dzielenie, pierwiastkowanie). Zamienia liczby binarne na ich reprezentację w pamięci maszyny liczącej w systemach ZM, U1, U2. Wykonuje operacje arytmetyczne na liczbach w systemach ZM, U1, U2. Zamienia liczby dziesiętne na system zmiennoprzecinkowy w standardzie IEEE 754 (zarówno w przypadku liczb znormalizowanych jak i zdenormalizowanych). Potrafi realizować operacje arytmetyczne na liczbach binarnych przy pomocy bramek logicznych.	K_U02
	U03: Przeprowadza generację i akceptację prostych języków formalnych.	K_U02
	U04: Potrafi zdefiniować i zbudować automat skończony i automat ze stosem. Rozwiązuje problemy algorytmiczne przy pomocy maszyny Turinga. Potrafi zapisać algorytmy i rozwiązywać je za pomocą Maszyny von Neumanna.	K_U02 K_U16
	U05: Potrafi tworzyć schematy blokowe dla algorytmów liniowych, rozgałęzionych oraz cyklicznych. Analizuje poprawność częściową i całkowitą prostego algorytmu; ocenia złożoność czasową i pamięciową przykładowych algorytmów i dokonuje ich optymalizacji.	K_U02 K_U16 K_U17 K_U18

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Kompetencje społeczne	K01: jest świadomy konieczności stałego uaktualniania wiedzy z zakresu informatyki, posiada zdolność do weryfikowania pozyskiwanych informacji, a także analizy podstawowych problemów teoretycznych informatyki .	K_K02 K_K01
	K02: jest świadomy konieczności dzielenia się wiedzą informatyczną w sposób zrozumiały dla innych.	K_K02 K_K01

Studia stacjonarne

Organizacja												
Forma zajęć	Wykład (W)	Ćwiczenia w grupach										
		A		K		L		S		P		E
Liczba godzin	25	30										

Studia niestacjonarne

Organizacja												
Forma zajęć	Wykład (W)	Ćwiczenia w grupach										
		A		K		L		S		P		E
Liczba godzin	15	15										

Opis metod prowadzenia zajęć

Omawianie zagadnień teoretycznych, prezentacje multimedialne, omawianie przypadków, rozwiązywanie zadań (ćwiczenia), wykonywanie projektów, referowanie zagadnień z użyciem środków multimedialnych.

Formy sprawdzania efektów uczenia się

	E – learning	Gry dydaktyczne	Ćwiczenia w szkole	Zajęcia terenowe	Praca laboratoryjna	Projekt indywidualny	Projekt grupowy	Udział w dyskusji	Referat	Praca pisemna (esej)	Egzamin ustny	Egzamin pisemny	Inne
W01						X			X	X		X	X
W02	X					X			X			X	X
W03						X			X	X		X	X
U01						X			X	X		X	
U02												X	
U03								X		X		X	
U04								X		X		X	
U05									X	X		X	
K01								X	X				
K02								X	X				

Kryteria oceny

Ocenę dobrą lub bardzo dobrą może uzyskać student, który:

- Ocena końcowa z ćwiczeń wystawiona będzie na podstawie ewaluacji ocen częściowych z kolokwii oraz ew. z przygotowanego i wygłoszonego referatu, jeśli takowy będzie miał miejsce.
- Bierze udział w dyskusji i zaangażowanie w rozwiązywaniu zadań w trakcie

ćwiczeń.

- Ocena przedmiotu będzie wystawiona w oparciu o ocenę z egzaminu testowego.

Uwagi

Treści merytoryczne (wykaz tematów)

- 1) Podstawy asemantycznej teorii informacji: ilościowy aspekt informacji, miara niepewności, podstawowe twierdzenie teorii informacji, jednostki pomiaru ilości informacji, niepewność maksymalna, redundancja i jej rola. Kompresja.
- 2) Teoria kodowania: łącze informacyjne, formalna definicja kodowania, podstawowe własności kodów, definicja sprawności kodu, sprawność a zwięzłość, metody kodowania Shannona, Fano i Huffmana. Metody zabezpieczania kodów, reguła parzystości, odległość Hamminga i jej znaczenie, kod paskowy. Systemy kodowania binarnego, oktalnego i heksadecymalnego liczb. Konwersje. Arytmetyka binarna.
- 3) Języki i gramatyki formalne: gramatyki generacyjne i redukcyjne, definicja formalna i opisowa gramatyki generacyjnej, rodzaje gramatyk generacyjnych a klasyfikacja języków formalnych. Hierarchia Chomsky'ego.
- 4) Automaty a akceptacja i translacja języków formalnych : rodzaje automatów i ich możliwości akceptacji przykładowych języków formalnych. Automaty z wyjściem a translacja. Parser i lexer jako elementy kompilatora. Model funkcjonalny kompilatora.
- 5) Języki i rodzaje i paradygmaty programowania
- 6) Modele maszyn cyfrowych:
maszyna Turinga: osprzęt i oprogramowanie, możliwości maszyny Turinga, uniwersalność maszyny Turinga, minimalna uniwersalna maszyna Turinga, przykłady zastosowań maszyny Turinga.
Przykładowa Maszyna Cyfrowa von Neumanna (PMC): struktura i schemat blokowy, struktura słowa maszynowego i jej konsekwencje. Sieć działań PMC. Przykładowy program w języku maszyny PMC.
- 7) Elementy architektury komputera:
Twierdzenie Posta i jego konsekwencje dla informatyki. Logika maszyny cyfrowej a algebra Boole'a. Bramki i układy logiczne. Budowa sumatora, koder a i dekoder
- 8) Elementy informatyki kwantowej. Komputery kwantowe.
- 9) Bioinformatyka i biokomputery
- 10) Elementy algorytmiki:
 - zadanie algorytmiczne, struktura algorytmów;
 - poprawność częściowa i całkowita algorytmów, metody dowodzenia (metoda Floyda i metoda zbieżnika) oraz przykłady zastosowań, współczesne tendencje w badaniu poprawności algorytmów, poprawność algorytmów a dowodzenie twierdzeń matematycznych (na przykładzie tw. o czterech barwach i tw. Keplera o upakowaniu kul)
 - złożoność czasowa i pamięciowa algorytmów, typy złożoności czasowej, optymalizacja algorytmów, problemy zamknięte i luka algorytmiczna, ograniczenia górne i dolne na złożoność czasową algorytmów, złożoność czasowa a złożoność pamięciowa
 - problemy obliczalności i rozstrzygalności algorytmów, przykłady algorytmów, problemy P, NP i NP-zupełne, teza Churcha-Turinga-Markowa i jej konsekwencje.
 - algorytmy kwantowe: algorytm Shora i algorytm Grovera a problem obliczalności i rozstrzygalności

Wykaz literatury podstawowej

Wskazane rozdziały publikacji:

1. N. Abramson „Teoria informacji i kodowania” ,PWN, Warszawa 1969.
N. Abramson “Information Theory and Coding”, Published by McGraw Hill, 1963.
2. W. Turski „Propedeutyka informatyki, PWN 1975
3. S. Kowalski i, A.W. Mostowski „ Teoria automatów i lingwistyka matematyczna”, PWN, Warszawa 1979.
4. J.E. Hopcroft , J.D. Ullman „ Wprowadzenie do teorii automatów, języków i obliczeń”, PWN, Warszawa 1994. „
5. K. Fijałkowski "Autokody i programowanie maszyn cyfrowych"; W-NT,Warszawa 1976 ,
6. W.H. Desmond, Maszyny matematyczne i ich zastosowania, PWN, Warszawa 1969
7. L. Banachowski, A. Kreczmar „Elementy analizy algorytmów”, W-NT, Warszawa 1982
6. D. Harel „ Rzecz o istocie informatyki – algorytmika ” WN-T, Warszawa 1992
7. M. Hirvensalo „Algorytmy kwantowe”, WSiP , Warszawa 2004
8. J. Bylina, B. Bylina „Przegląd języków i paradygmatów programowania”, UMCS, Lublin 2011

Wykaz literatury uzupełniającej

1. I. Pohl and A Shaw, „The nature of computation: an introduction to computer science”, Computer Science Press Inc., MD USA 1981
2. J. Glen Brookshear „Informatyka w ogólnym zarysie” W N-T Warszawa 2003.
3. M.M. Sysło „Algorytmy”, WSiP, Warszawa 2002
5. R. Tadeusiewicz, P. Moszner, A. Szydełko „Teoretyczne podstawy informatyki”, WN WSP, Kraków 1998
6. W. Duch :”Fascynujący świat programów komputerowych”, Internet
7. Materiały dydaktyczne opracowane przez prowadzącego kurs w formie opisowej i prezentacji zamieszczone na platformie e-learningowej Moodle.
8. MacKay David J.C. Information Theory, Inference, and Learning Algorithms, Cambridge University Press, 2003.
9. Information and Coding Theory, Gareth A. Jones and J. Mary Jones, Springer, 2000.
10. Elements of Information Theory, Thomas M. Cover and Joy A. Thomas, Wiley Series in Telecommunications, 1991.

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) – studia stacjonarne

Liczba godzin w kontakcie z prowadzącymi	Wykład	25
	Konwersatorium (ćwiczenia, laboratorium itd.)	30
	Pozostałe godziny kontaktu studenta z prowadzącym	15
Liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	15
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	15
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	20
	Przygotowanie do egzaminu/zaliczenia	30
Ogółem bilans czasu pracy		150
Liczba punktów ECTS w zależności od przyjętego przelicznika		6

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) – studia niestacjonarne

Liczba godzin w kontakcie z prowadzącymi	Wykład	15
	Konwersatorium (ćwiczenia, laboratorium itd.)	15
	Pozostałe godziny kontaktu studenta z prowadzącym	10
Liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	35
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	20
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	20
	Przygotowanie do egzaminu/zaliczenia	35
Ogółem bilans czasu pracy		150
Liczba punktów ECTS w zależności od przyjętego przelicznika		6