

KARTA KURSU

Nazwa	Teoretyczne podstawy informatyki
Nazwa w j. ang.	Theoretical Foundations of Computer Science

Koordynator	dr hab. Tomasz Dobrowolski, prof. Uken	Zespół dydaktyczny
		dr Iryna Artyschchuk mgr Justyna Golec
Punktacja ECTS*	st. stacjonarne: 4 st. niestacjonarne: 4	dr hab. Tomasz Dobrowolski, prof. Uken

Opis kursu (cele kształcenia)

Celem realizacji kursu jest zapoznanie studentów kierunku Informatyka z podstawami informatyki oraz możliwościami wynikającymi z wykształcenia informatycznego oraz potencjalnych kierunków dalszego rozwoju. Student powinien uzyskać wiedzę na temat pozycyjnych systemów liczbowych oraz ich implementacji w pamięci maszyny liczącej. Zapoznać się z pojęciem oraz głównymi cechami algorytmów. Zyskać wiedzę na temat struktur danych. Pozyskać wiadomości na temat języków formalnych, wyrażeń regularnych, automatów skończonych oraz gramatyk bezkontekstowych. Zapoznać się elementarnymi wiadomościami na temat teorii obliczalności. Uzyskać elementarną wiedzę na temat teorii informacji. Celem kursu jest przygotowanie studentów do samodzielnego uczenia się i uzupełniania wiedzy w zakresie podstaw informatyki, a także możliwościami wynikającymi z wykształcenia informatycznego oraz potencjalnymi kierunkami dalszego rozwoju.

Kurs prowadzony jest w języku polskim.

Warunki wstępne

Wiedza	Podstawowe definicje i pojęcia matematyki i informatyki szkolnej.
Umiejętności	Umiejętność korzystania z komputera, oprogramowania MS Office, Internetu.
Kursy	

Efekty uczenia się

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Wiedza	Po zakończeniu kursu student: W01: ma wiedzę na temat pozycyjnych systemów liczbowych, W02: zna główne typy algorytmów, poznał podstawowe struktury danych, W03: uzyskał wiedzę na temat języków formalnych, wyrażeń regularnych, automatów skończonych oraz gramatyk bezkontekstowych, zna podstawowe pojęcia teorii obliczalności,	K_W03 K_W08

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Umiejętności	<p>Po zakończeniu kursu student:</p> <p>U01: 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 reprezentacje 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).</p> <p>U02: 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.</p> <p>U03: Przeprowadza generację i akceptację prostych języków formalnych</p> <p>U04: Potrafi zdefiniować i zbudować automat skończony i automat ze stosem.</p>	<p>K_U11 K_U12 K_U13 K_U14</p>

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

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	20									

Opis metod prowadzenia zajęć

1. Wykłady: Podczas wykładów wykładowcy wprowadzają materiał teoretyczny, wyjaśniają kluczowe pojęcia i metody oraz przedstawiają przykłady i ilustracje. Wykłady mogą być prowadzone stacjonarnie lub online.
2. Dyskusje grupowe i zadania: dyskusje grupowe i zadania ułatwiają dzielenie się wiedzą między studentami i zachęcają do wspólnego uczenia się.
3. Samodzielna nauka: Studenci mogą również mieć dostęp do materiałów do samodzielnej nauki, takich jak podręczniki, artykuły i kursy online. Pozwala to uczniom na pogłębienie wiedzy i zbadanie tematów, które ich szczególnie interesują.

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
W03						X			X	X		X	X
W03						X			X	X		X	X
U01						X		X	X	X		X	
U02						X		X				X	
U03						X		X		X		X	
U04						X		X		X		X	
K01								X	X				
K02								X	X				

Kryteria oceny

Zaliczenie ćwiczeń zależy od ocen cząstkowych oraz regularności wykonywania zadań. Ocena końcowa zależy od aktywności w czasie ćwiczeń oraz oceny otrzymanej na egzaminie końcowym.

Uwagi

Treści merytoryczne (wykaz tematów)

1. Wprowadzenie do informatyki
Definicja i krótka historia informatyki
2. Rola informatyki we współczesnym świecie
Zawody i role w branży IT
Przegląd poszukiwanych zawodów (programista, administrator systemów, analityk danych, specjalista ds. cyberbezpieczeństwa itp.)
Krótki opis obowiązków i wymaganych umiejętności dla każdej roli
3. Systemy liczbowe
Pozycyjne systemy liczbowania (dziesiętny, binarny, ósemkowy, szesnastkowy)
Konwersja liczb z jednego systemu liczbowego na inny
Podstawowe operacje w różnych systemach liczbowych
4. Podstawy algorytmów i programowania
Definicja i główne cechy algorytmów
5. Pojęcie zmiennych, typów danych i operatorów
Podstawowe konstrukcje sterujące (instrukcje warunkowe, pętle)
Przykłady prostych algorytmów (sortowanie, wyszukiwanie)
6. Podstawy struktur danych
Wprowadzenie do tablic, list, stosów i kolejek
Przykłady użycia struktur danych

7. Języki formalne i gramatyki
 Podstawowe pojęcia języków formalnych
 Wyrażenia regularne i automaty skończone
 Gramatyki bezkontekstowe (podstawowe pojęcia i przykłady)
 7. Teoria obliczalności
 Wprowadzenie do maszyn Turinga (podstawowe koncepcje)
 Pojęcie obliczalności i nieobliczalności
 Problem stopu (wprowadzenie)
 8. Podstawowe wiadomości z teorii informacji.
 9. Kod ASCII i jego rozszerzenia.
 10. Współczesne trendy w IT
 Obliczenia w chmurze
 Sztuczna inteligencja i uczenie maszynowe
 Big data i ich analiza
 Internet rzeczy (IoT)

Wykaz literatury podstawowej

1. Pochopień B., Stańczyk U., Wróbel E., Arytmetyka systemów cyfrowych w teorii i praktyce, ISBN: 978-83-7335-988-8.
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.
8. D. Harel „Rzecz o istocie informatyki – algorytmika” WN-T, Warszawa 1992.

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”.
7. MacKay David J.C. Information Theory, Inference, and Learning Algorithms, Cambridge University Press, 2003.

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	5
Liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	10
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	10
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	10
	Przygotowanie do egzaminu/zaliczenia	10
Ogółem bilans czasu pracy		100
Liczba punktów ECTS w zależności od przyjętego przelicznika		4

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

Liczba godzin w kontakcie z prowadzącymi	Wykład	15
	Konwersatorium (ćwiczenia, laboratorium itd.)	20
	Pozostałe godziny kontaktu studenta z prowadzącym	10
Liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	20
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	10
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	15
	Przygotowanie do egzaminu/zaliczenia	10
Ogółem bilans czasu pracy		100
Liczba punktów ECTS w zależności od przyjętego przelicznika		4