



IV. Opis programu studiów

3. KARTA PRZEDMIOTU

Kod przedmiotu	E-I-0593-s1
Nazwa przedmiotu	Wstęp do informatyki
Nazwa przedmiotu w języku angielskim	Introduction to computer science
Obowiązuje od roku akademickiego	2019/20

USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	
Poziom kształcenia	
Profil studiów	
Forma i tryb prowadzenia studiów	
Zakres	
Jednostka prowadząca przedmiot	
Koordinator przedmiotu	dr inż. Remigiusz Baran
Zatwierdził	Dziekan Wydziału Elektrotechniki Automatyki i Informatyki Dr hab. inż. Antoni Różowicz, prof. PŚk

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	
Status przedmiotu	
Język prowadzenia zajęć	
Usytuowanie modułu w planie studiów - semestr	
Wymagania wstępne	
Egzamin (TAK/NIE)	
Liczba punktów ECTS	

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	30	0	15	0	0

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Ma elementarną wiedzę dotyczącą fundamentalnych pojęć informatyki, w tym systemów liczbowych oraz technik przetwarzania informacji z wykorzystaniem podstawowych i złożonych typów danych.	INF1_W06 INF1_W07
	W02	Rozumie pojęcie algorytmu, przyswoił zagadnienia z zakresu projektowania algorytmów, poznał typowe zadania algorytmiczne i strategie ich rozwiązywania.	INF1_W06 INF1_W07
	W03	Zna i rozumie proceduralny język programowania. Orientuje się w ważnych, wybranych zagadnieniach informatyki obejmujących m.in. paradygmaty i języki programowania, kryptografię, sztuczną inteligencję i programowanie współbieżne.	INF1_W13
Umiejętności	U01	Opanował operacje arytmetyczne i bitowe na danych, z wykorzystaniem pozycyjnych systemów liczbowych.	INF1_U15
	U02	Umie projektować algorytmy metodą graficzną i analizować ich złożoność obliczeniową. Potrafi zapisywać algorytmy w proceduralnym języku programowania.	INF1_U14
	U03	Potrafi stosować proste i strukturalne typy danych. Potrafi zaprojektować strukturę złożonego typu danych.	INF1_U15
Kompetencje społeczne	K01	Posiada świadomość wpływu wybranych narzędzi z obszaru informatyki na rozwój innych dziedzin nauki.	INF1_K02

TREŚCI PROGRAMOWE

Forma zajęć*	Treści programowe
wykład	1. Pojęcia podstawowe. Informatyka. Wprowadzenie do teorii informacji. Miary ilości informacji. Modele danych - modele Markowa. Elementy logiki i algebry Boole'a.
	2. Arytmetyka binarna i systemy liczbowe Arytmetyka binarna w kodowaniach NKB, U2 i BCD. Pozycyjne systemy liczbowe. Konwersje do dziesiętnych i niedziesiętnych systemów liczbowych. Stało- i zmiennopozycyjne reprezentacje danych.
	3. Język proceduralny Architektura von Neumana. Architektura prostego komputera. Cykl rozkazowy procesora. Instrukcje kodu maszynowego. Języki niskiego i wysokiego poziomu. Mechanizmy języka proceduralnego. Konstrukcje strukturalne i ich realizacja w kodzie maszynowym.
	4. Podstawy algorytmiki Pojęcia zmiennej, tablicy danych, algorytmu. Metody zapisu (dokumentowania) algorytmu. Strukturalne schematy blokowe i usuwanie niestrukturalności. Pseudokod.

	5. Podstawowe typy danych. Typy danych reprezentujące liczby całkowite i rzeczywiste w wybranych językach programowania. Logiczne typy danych. Systemy kodowania znaków.
	6. Elementarne operacje z wykorzystaniem podstawowych typów danych. Operacje arytmetyczne, logiczne i bitowe z wykorzystaniem liczb całkowitych o ograniczonej długości bitowej. Arytmetyka stało- i zmiennopozycyjna.
	7. Złożone typy danych. Tablice jedno- i dwuwymiarowe liczb całkowitych/rzeczywistych (wektory i macierze). Adresowanie elementów tablic. Łańcuchy znaków. Struktury (rekordy) i unie.
	8. Podstawowych konstrukcje algorytmiczne. Algorytmy przeszukiwania enumeratywnego, przetwarzające dane zawarte w tablicach jedno- lub dwuwymiarowych (sumowanie wartości elementów, wyznaczanie elementu o maksymalnej/minimalnej wartości, itp.). Podstawowe algorytmy sortowania.
	9. Procedury i funkcje. Podejścia zstępujące i wstępujące w programowaniu. Przekazywanie parametrów do procedur i funkcji. Stos systemowy.
	10. Rekurencja Mechanizm rekurencji. Problemy rozwiązywane z użyciem rekurencji. Implementacja rekurencji.
	11. Złożoność obliczeniowa algorytmów. Pojęcie złożoności obliczeniowej czasowej i pamięciowej. Asymptotyczne tempo wzrostu. Notacja O. Maszyna Turinga. Algorytmy deterministyczne i niedeterministyczne.
	12. Języki i paradygmaty programowania. Podstawowe paradygmaty programowania. Wybrane języki programowania.
laboratorium	13. Inne zagadnienia Informatyki Elementy kryptografii. Wprowadzenie do programowania współbieżnego. Sztuczna inteligencja.
	1. Wprowadzenie do zajęć laboratoryjnych. Zapoznanie z wybranym środowiskiem programistycznym oraz z dokumentacją dotyczącą zajęć laboratoryjnych.
	2. Operacje arytmetyczne w wybranych systemach liczbowych. Operacje arytmetyczne w pozycyjnych systemach liczbowych NKB i U2.
	3. Operacje bitowe w wybranych systemach liczbowych. Operacje bitowe w pozycyjnych systemach liczbowych NKB i U2.
	4. Projektowanie algorytmów metodą graficzną. Projektowanie metodą graficzną wybranych algorytmów enumeratywnych.
	5. Implementacje algorytmów. Implementacje algorytmów z wykorzystaniem wybranych języków niskiego oraz wysokiego poziomu.
	6. Złożone typy danych. Badanie reprezentacji binarnej tablic jedno- i dwuwymiarowych, struktur oraz unii danych.

*) zostawić tylko realizowane formy zajęć

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów kształcenia					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01		X				
W02		X				
W03		X				
U01			X		X	
U02			X		X	
U03			X		X	
K01		X				

A.**FORMA I WARUNKI ZALICZENIA**

Forma zajęć*	Forma zaliczenia	Warunki zaliczenia
wykład		Uzyskanie co najmniej 50% punktów z egzaminu pisemnego.
laboratorium		Uzyskanie co najmniej 50% punktów z kolokwium w trakcie zajęć. Pozytywna ocena ze sprawozdań.

*) zostawić tylko realizowane formy zajęć

NAKŁAD PRACY STUDENTA

Bilans punktów ECTS							
Lp.	Rodzaj aktywności	Obciążenie studenta					Jednostka
1.	Udział w zajęciach zgodnie z planem studiów	W	C	L	P	S	h
		30		15			
3.	Inne (konsultacje, egzamin)*	3					h
4.	Razem przy bezpośrednim udziale nauczyciela akademickiego	48					h
5.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	1,92					ECTS
6.	Liczba godzin samodzielnej pracy studenta	52					h
7.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	2,08					ECTS
8.	Nakład pracy związany z zajęciami o charakterze praktycznym	15					h
9.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	1,25					ECTS
10.	Sumaryczne godzinowe obciążenie pracą studenta	100					h
11.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	4					

* wszelkie formy weryfikacji efektów, w tym egzaminy oraz nie więcej niż 2 godziny konsultacji dla każdej formy zajęć

LITERATURA

1. Harel David, Rzecz o istocie informatyki. Algorytmika, Seria: Klasyka informatyki, WNT, Wyd IV, 2008
2. Alfred V. Aho, Jeffrey D. Ullman, Wykłady z informatyki z przykładami w języku C, Helion, 2003
3. Alfred V. Aho, Jeffrey D. Ullman, Foundations of Computer Science: C Edition, Principles of Computer Science Series, 1994, <http://infolab.stanford.edu/~ullman/focs.html>
4. Aho A.J., Hopcroft J.E., Ullman J.D.: "Struktury danych i algorytmy", PWN, 1983.
5. Cormen T., Leiserson C., Rivest R.: "Wprowadzenie do algorytmów", WNT, 1997.
6. Dembiński P., Małuszyński J.: "Matematyczne metody definiowania języków programowania", WNT, Warszawa 1981.
7. Wirth N.: "Algorytmy + struktury danych = programy", WNT, 1989.
8. Stallings W., Organizacja i architektura systemu komputerowego, WNT, 2009

Uwaga: wykaz literatury winien uwzględniać aktualne i dostępne publikacje