

**KAPITAŁ LUDZKI**
NARODOWA STRATEGIA SPÓJNOŚCIProjekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego**UNIA EUROPEJSKA**
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Nazwa przedmiotu		Kod ECTS	
Wstęp do programowania		11.3.0722	
Nazwa jednostki prowadzącej przedmiot			
Faculty of Mathematics, Physics and Informatics			
Studia			
wydział	kierunek	poziom	pierwszego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
dr Janusz Dybizbański; dr Piotr Arłukowicz; dr Andrzej Borzyszkowski; dr Paweł Pączkowski; mgr Mateusz Miotk; mgr Omer Sakarya; dr Karol Horodecki; dr Magdalena Godlewska; dr Marcin Ciecholewski; mgr inż. Anna Nenca; prof. UG, dr hab. Tomasz Dzido; prof. UG, dr hab. Marcin Wieśniak; dr inż. Jerzy Skurczyński; mgr Gabriela Łuczyńska; mgr Adam Kostulak; mgr Radosław Ziemann; dr Mikołaj Czechlewski; mgr Maciej Dziemiańczuk; mgr inż. Anna Nenca			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		8	
Wykład, Ćw. audytoryjne, Ćw. laboratoryjne		Przedmiot w wymiarze 30h wykładu, 30h ćw. aud. i 30h ćw. laboratoryjnych + praca własna studenta	
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Ćw. laboratoryjne: 30 godz., Ćw. audytoryjne: 30 godz., Wykład: 30 godz.			
Cykl dydaktyczny			
2015/2016 zimowy			
Status przedmiotu		Język wykładowy	
obowiązkowy		polski	
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne	
<ul style="list-style-type: none"> - Rozwiązywanie zadań - Wykład z prezentacją multimedialną - ćwiczenia laboratoryjne -- sporządzanie i uruchamianie programów komputerowych 		Sposób zaliczenia	
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - egzamin pisemny z pytaniami (zadaniami) otwartymi - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru - kolokwium - egzamin pisemny (dłuższa wypowiedź pisemna / rozwiązanie problemu) 	
		Podstawowe kryteria oceny	
		Przedmiot kończy się egzaminem z pytaniami otwartymi. Warunkiem przystąpienia do egzaminu jest uzyskanie pozytywnych ocen zarówno z ćwiczeń laboratoryjnych jak i ćwiczeń audytoryjnych.	
Sposób weryfikacji założonych efektów kształcenia			

zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność w dyskusji	obserwacja i ocena postawy i umiejętności studenta
Wiedza							
K_W02	X	X	X			X	X
K_W03	X	X	X			X	X
K_W04	X	X	X			X	X
Umiejętności							
K_U01	X	X	X			X	X
K_U03	X	X	X			X	X
K_U05			X			X	X
K_U06	X	X	X			X	X
K_U08	X	X	X			X	X
Kompetencje							
K_K02						X	X

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

A. Wymagania formalne

B. Wymagania wstępne

Cele kształcenia

Celem przedmiotu jest poznanie podstawowych technik programistycznych oraz zdobycie umiejętności projektowania, analizy i implementacji podstawowych algorytmów.

Treści programowe

- Algorytm a program. Proste programy w różnych sposobach zapisu (opis słowny, schemat blokowy, instrukcje języka programowania). Ręczna symulacja działania algorytmu.
- Schemat przetwarzania programu w języku wysokiego poziomu - etapy kompilacji, konsolidacji i wykonania.
- Deklaracje zmiennych. Proste typy danych oraz typy strukturalne: tablica, rekord. Zakres deklaracji i widoczności zmiennych, zmienne globalne.
- Instrukcje sterujące. Pojęcie poprawności częściowej i całkowitej programu. Dowodzenie poprawności metodą niezmienników. Zagnieżdżone pętle.
- Procedury i funkcje. Sposoby przekazywania parametrów do funkcji.
- Mechanizm rekursji i jego wykorzystanie. Wyrażanie pętli przez rekursję i przykład wyrażenia rekursji pętlą. Dowodzenie poprawności funkcji rekurencyjnych za pomocą indukcji matematycznej.
- Szacowanie liczby operacji wykonywanych przez algorytm.
- Reprezentacja liczb całkowitych i zmiennoprzecinkowych w komputerze oraz błędy nimi spowodowane (przekroczenie zakresu i błędy zaokrągleń).
- Wykorzystanie wskaźników i zarządzanie pamięcią na przykładzie prostych struktur danych.
- Informacja o teoretycznych ograniczeniach obliczeniowych.

Wykaz literatury

- [1] Griffiths David, Griffiths Dawn. *C. Rusz głową!* Wydawnictwo Helion.
 [2] Szepietowski A. *Podstawy informatyki*. Wydawnictwo Uniwersytetu Gdańskiego, Gdańsk 2000.
 [3] Kernighan B.W., Ritchie D.M. *Język ANSI C*. Wydawnictwa Naukowo-Techniczne, Warszawa 2003.

Efekty kształcenia (obszarowe i kierunkowe)

K_W02 ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie programowania, algorytmów i złożoności, architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, języków i paradygmatów programowania, baz danych, inżynierii oprogramowania, języków formalnych,
 K_W03 zna podstawowe metody projektowania, analizowania i programowania algorytmów,

Wiedza

- Student:
- zna podstawowe narzędzia programistyczne
 - zna elementy programów komputerowych: zmienne, instrukcje proste i sterujące, funkcje
 - ma podstawy wiedzy o sposobach analizy poprawności i złożoności algorytmów
 - zna typowe algorytmów obliczania wyszukiwania i porządkowania
 - zna mechanizm rekursji
 - zna sposoby reprezentacji liczb w komputerze
 - zna podstawy technik programistycznych: dziel i rządz, DP

<p>K_W04 zna podstawowe konstrukcje programistyczne oraz struktury danych,</p> <p>K_U01 potrafi zastosować wiedzę matematyczną do formułowania, analizowania i rozwiązywania prostych zadań związanych z informatyką,</p> <p>K_U03 potrafi pracować indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów,</p> <p>K_U05 potrafi pisać, uruchamiać i testować programy w wybranym środowisku programistycznym</p> <p>K_U06 projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy; wykorzystuje podstawowe techniki algorytmiczne i struktur danych</p> <p>K_U08 posługuje się przyjętymi formatami reprezentacji różnego rodzaju danych stosownie do sytuacji</p> <p>K_K02 potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania</p>	<p>- zna podstawowe struktury danych: stos, kolejka</p>
	<p>Umiejętności</p> <p>Student:</p> <ul style="list-style-type: none"> - umie przeprowadzić ręczną symulacja działania prostego programu z pętlami lub rekursją - umie zaprojektować i zaimplementować program z pętlami (również zagnieżdżonymi) oraz funkcjami - umie uzasadnić, że napisany program działa zgodnie z (nieformalnymi) wymaganiami - potrafi ocenić liczbę operacji, wykonywanych przez program z pętlami - umie zaprojektować proste funkcje rekurencyjne - potrafi zapisywać i odczytywać liczby jako ciągu bitów, zarówno w przypadku stało- jak i zmienna-pozycyjnym
	<p>Kompetencje społeczne (postawy)</p> <p>Student:</p> <ul style="list-style-type: none"> - potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania
<p>Kontakt</p> <p>jdybiz@inf.ug.edu.pl</p>	