



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Nazwa przedmiotu		Kod ECTS						
Języki programowania		11.3.1316						
Nazwa jednostki prowadzącej przedmiot								
Instytut Informatyki								
Studia								
wydział	kierunek	poziom	pierwszego stopnia					
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	niestacjonarne (zaoczne)					
		moduł	wszystkie					
		specjalnościowy	wszystkie					
		specjalizacja	wszystkie					
Nazwisko osoby prowadzącej (osób prowadzących)								
dr Maciej Dziemiańczuk; dr Adam Kostulak; mgr Łukasz Mielewczyk; dr Andrzej Borzyszkowski; dr inż. Arkadiusz Mirakowski; dr hab. Tomasz Dzido; mgr Mateusz Miotk								
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS						
Formy zajęć		12 Przedmiot w wymiarze 40h wykładu i 40h lab. + praca własna studenta						
Wykład, Ćw. laboratoryjne								
Sposób realizacji zajęć								
zajęcia w sali dydaktycznej								
Liczba godzin								
Wykład: 40 godz., Ćw. laboratoryjne: 40 godz.								
Termin realizacji przedmiotu								
2020/2021 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"> - Wykład z prezentacją multimedialną - ćwiczenia na pracowni komputerowej – rozwiązywanie zadań 		Sposób zaliczenia						
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 						
		Formy zaliczenia						
		egzamin pisemny testowy						
		Podstawowe kryteria oceny						
		Ocena z ćwiczeń na podstawie wyników z prac domowych oraz aktywności na zajęciach. Ocena z wykładu na podstawie wyników testu sprawdzającego rozumienie fragmentów kodu.						
Sposób weryfikacji założonych efektów kształcenia								
zakładany efekt kształcenia	egzamin	kolokwium	projekt	sprawdzian	referat	raport	aktywność w dyskusji	obserwacja
	Wiedza							
K_W05	X							
	Umiejętności							
K_U07								X
K_U10								X
Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi								
A. Wymagania formalne								
Brak.								

<p>B. Wymagania wstępne Umiejętność obsługi komputera i instalacji programów.</p>	
<p>Cele kształcenia Celem jest nauczenie studentów programowania w języku C oraz czytania ze zrozumieniem kodu C napisanego przez innych programistów.</p>	
<p>Treści programowe</p> <ol style="list-style-type: none"> 1. <i>Wprowadzenie do programowania w języku C.</i> Uruchamianie prostych programów. Składnia i semantyka programu. 2. <i>Środowisko programistyczne w systemie UNIX.</i> Kompilacja i konsolidacja programu z użyciem kompilatora <i>gcc</i>. Program <i>make</i>. Debugger <i>gdb</i>. 3. <i>Typy, operatory i wyrażenia.</i> Deklaracje. Konwersja typów. Wyrażenia warunkowe. Kolejność obliczeń. 4. <i>Instrukcje sterujące.</i> Instrukcje i bloki. 5. <i>Funkcje i struktura programu.</i> Zasięg nazw. Pliki nagłówkowe. Preprocesor języka C. Kompilacja warunkowa. 6. <i>Wskaźniki i tablice.</i> Argumenty wywołania programu. Wskaźniki do funkcji. Skomplikowane deklaracje. 7. <i>Struktury.</i> Deklaracje <i>typedef</i>. Unie. 8. <i>Wejścia i wyjścia.</i> Omówienie standardowych bibliotek: wejścia/wyjścia, funkcji działających na napisach, funkcji matematycznych. Wprowadzenie do tzw. locales. Kodowanie UTF-8. 9. <i>Abstrakcyjne typy danych i ich implementacja.</i> Stosy i kolejki. Drzewa poszukiwań binarnych. Tablice dynamiczne. 	
<p>Wykaz literatury</p> <ol style="list-style-type: none"> 1. Brian W. Kernighan, Dennis M. Ritchie. Język ANSI C. WNT 2. Stephen Prata. Język C. Szkoła programowania, Helion 3. Keith Cooper, Linda Torczon. Engineering a Compiler, MORGAN KAUFMANN <p>Podręczniki online:</p> <ol style="list-style-type: none"> 1. Zed A. Shaw. Learn C The Hard Way 2. Mike Banahan, Declan Brady, Mark Doran. The C Book 3. Richard M. Stallman, Roland McGrath, Paul D. Smith. GNU Make Manual 4. Sandra Loosemore i Richard M. Stallman, Roland McGrath, Andrew Oram, Ulrich Drepper. The GNU C Library 	
<p>Kierunkowe efekty kształcenia</p> <p>K_W05:ma ogólną wiedzę na temat różnych paradygmatów programowania i języków programowania; szczególnie zna metody i wzorce projektowania i programowania obiektowego</p> <p>K_U07:potrafi projektować, tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz adekwatnych wzorców</p> <p>K_U10:potrafi oceniać przydatność paradygmatów i narzędzi programistycznych do rozwiązywania problemów różnego typu</p>	<p>Wiedza</p> <p>Odróżnia błędy składniowe w kodzie programów od błędów logicznych. Rozumie krótkie fragmenty kodu. Zna podstawowe konstrukcje języka C. Wie jakie są biblioteki standardowe języka i co zawierają.</p>
	<p>Umiejętności</p> <p>Student:</p> <ul style="list-style-type: none"> • umie napisać program w C rozwiązujący konkretny problem • umie zaprogramować w C prosty algorytm • w swoich programach potrafi skorzystać z gotowych bibliotek • potrafi uruchamiać programy • potrafi zaadaptować cudzy kod do swoich potrzeb
	<p>Kompetencje społeczne (postawy)</p> <p>Student rozumie potrzebę dalszego kształcenia.</p>
<p>Kontakt maciej.dziemianczuk@inf.ug.edu.pl</p>	