


**KAPITAŁ LUDZKI**  
 NARODOWA STRATEGIA SPÓJNOŚCI

 Projekt współfinansowany przez  
 Unię Europejską w ramach  
 Europejskiego Funduszu  
 Społecznego

**UNIA EUROPEJSKA**  
 EUROPEJSKI  
 FUNDUSZ SPOŁECZNY


<b>Nazwa przedmiotu</b>		<b>Kod ECTS</b>	
Wstęp do programowania		11.3.1512	
<b>Nazwa jednostki prowadzącej przedmiot</b>			
Instytut Informatyki			
<b>Studia</b>			
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>pierwszego stopnia</b>
Wydział Matematyki, Fizyki i Informatyki	Informatyka	<b>forma</b>	stacjonarne
		<b>moduł</b>	wszystkie
		<b>specjalnościowy</b>	wszystkie
		<b>specjalizacja</b>	wszystkie
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>			
dr Janusz Dybizbański; dr inż. Anna Nenca; prof. UG, dr hab. Marcin Ciecholewski			
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>	
<b>Formy zajęć</b>		7 Przedmiot w wymiarze 30h wykładu, 45h ćw. laboratoryjnych + praca własna studenta	
Wykład, Ćw. laboratoryjne			
<b>Sposób realizacji zajęć</b>			
zajęcia w sali dydaktycznej			
<b>Liczba godzin</b>			
Ćw. laboratoryjne: 45 godz., Wykład: 30 godz.			
<b>Termin realizacji przedmiotu</b>			
2022/2023 zimowy			
<b>Status przedmiotu</b>		<b>Język wykładowy</b>	
obowiązkowy		polski	
<b>Metody dydaktyczne</b>		<b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>	
<ul style="list-style-type: none"> <li>- Wykład z prezentacją multimedialną</li> <li>- ćwiczenia laboratoryjne -- sporządzanie i uruchamianie programów komputerowych</li> </ul>		<b>Sposób zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Egzamin</li> </ul>	
		<b>Formy zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- egzamin pisemny z pytaniami (zadaniami) otwartymi</li> <li>- ustalenie oceny zaliczeniowej na podstawie ocen częściowych otrzymywanych w trakcie trwania semestru</li> <li>- egzamin pisemny (dłuższa wypowiedź pisemna / rozwiązanie problemu)</li> <li>- kolokwium</li> </ul>	
		<b>Podstawowe kryteria oceny</b>	
		Przedmiot kończy się egzaminem z pytaniami otwartymi. Warunkiem przystąpienia do egzaminu jest uzyskanie pozytywnych ocen zarówno z ćwiczeń laboratoryjnych jak i ćwiczeń audytoryjnych.	
<b>Sposób weryfikacji założonych efektów uczenia się</b>			

zakładany efekt kształcenia	egzamin	kolokwium	projekt	sprawdzian	referat	raport	aktywność w dyskusji	obserwacja postawy
Wiedza								
K_W05	X	X	X					
K_W10								X
P_W1	X	X	X					
P_W2	X	X	X					
P_W3								X
Umiejętności								
K_U02							X	X
K_U06		X						X
P_U1		X					X	X
P_U2							X	X
P_U3		X					X	X
Kompetencje								
K_K01							X	X
K_K03							X	X
P_K1							X	X
P_K2							X	X

**Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi****A. Wymagania formalne**

brak wymagań formalnych

**B. Wymagania wstępne**

brak wymagań wstępnych

**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 pętłą. 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 zaokrąglenia).
- 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.

**Kierunkowe efekty uczenia się**

K\_W05: ma ogólną wiedzę na temat różnych

**Wiedza**

Student:

<p>paradygmatów programowania i języków programowania; szczególnie zna metody i wzorce projektowania i programowania obiektowego</p> <p>K_W10: zna podstawowe zasady bezpieczeństwa i higieny pracy w zawodzie informatyka</p> <p>K_U02: potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania</p> <p>K_U06: potrafi projektować, tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz adekwatnych wzorców</p> <p>K_K01: zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się</p> <p>K_K03: potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych</p>	<ul style="list-style-type: none"> <li>- zna podstawowe narzędzia programistyczne</li> <li>- zna elementy programów komputerowych: zmienne, instrukcje proste i sterujące, funkcje</li> <li>- ma podstawy wiedzy o sposobach analizy poprawności i złożoności algorytmów</li> <li>- zna typowe algorytmy wyszukiwania i porządkowania</li> <li>- zna mechanizm rekursji</li> <li>- zna sposoby reprezentacji liczb w komputerze</li> <li>- zna podstawy technik programistycznych: dziel i rządź, programowanie dynamiczne</li> <li>- zna podstawowe struktury danych: stos, kolejka</li> </ul> <p>Efekty przedmiotowe:</p> <p>P_W1 student umie tworzyć programy komputerowe i zna podstawowe techniki projektowania algorytmów (K_W05)</p> <p>P_W2 student zna mechanizm rekursji oraz podstawowe algorytmy i struktury danych (K_W05)</p> <p>P_W3 student zna podstawowe zasady korzystania z sieci i stanowisk komputerowych (K_W10)</p>
	<p><b>Umiejętności</b></p> <p>Student:</p> <ul style="list-style-type: none"> <li>- umie przeprowadzić ręczną symulację działania prostego programu z pętlami lub rekursją</li> <li>- umie zaprojektować i zaimplementować program z pętlami (również zagnieżdżonymi) oraz funkcjami</li> <li>- umie uzasadnić, że napisany program działa zgodnie z (nieformalnymi) wymaganiami</li> <li>- potrafi ocenić liczbę operacji, wykonywanych przez program z pętlami</li> <li>- umie zaprojektować proste funkcje rekurencyjne</li> <li>- potrafi zapisywać i odczytywać liczby jako ciągi bitów, zarówno w przypadku stałojak i zmiennopozycyjnym</li> </ul> <p>Efekty przedmiotowe:</p> <p>P_U1 student umie wyszukiwać i poprawiać błędy w istniejących programach komputerowych (K_U02, K_U06)</p> <p>P_U2 student umie ocenić i uzasadnić optymalność rozwiązań prostych problemów (K_U02)</p> <p>P_U3 student umie projektować proste algorytmy, w tym korzystające z techniki DP oraz 'dziel i zwyciężaj' (K_U06)</p>
	<p><b>Kompetencje społeczne (postawy)</b></p> <p>Student:</p> <ul style="list-style-type: none"> <li>- potrafi formułować wymagania dotyczące algorytmów</li> <li>- rozumie konieczność dalszego kształcenia się</li> </ul> <p>Efekty przedmiotowe:</p> <p>P_K1: student umie formułować opinie na temat podstawowych algorytmów i ograniczeń języków programowania (K_K01)</p> <p>P_K2: student rozumie konieczność rozwijania swojej wiedzy (K_K03)</p>
<p><b>Kontakt</b></p> <p>jdybiz@inf.ug.edu.pl</p>	