



**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  |                 | 11.3.1106  |                           |
| <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,<br>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; mgr Mateusz Miotk; dr Piotr Arłukowicz; prof. UG, dr hab. Tomasz Dzido; dr inż. Jerzy Skurczyński; dr hab. Marcin Ciecholewski; prof. UG, dr hab. Marcin Wieśniak; dr Karol Horodecki; dr Adam Kostulak; dr Paweł Pączkowski; mgr Radosław Ziemann; dr inż. Arkadiusz Mirakowski; mgr Omer Sakarya; mgr inż. Anna Nenca; dr Mikołaj Czechlewski; mgr Gabriela Łuczyńska; dr Magdalena Godlewska; dr inż. Emilia Lubecka; mgr Łukasz Mielewczyk; dr Andrzej Borzyszkowski; mgr inż. Anna Nenca; mgr Maciej Dziemiańczuk |                 |  |                           |
| <b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>   |                 | <b>Liczba punktów ECTS</b>   |                           |
| <b>Formy zajęć</b>  |                 | 8<br>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>   |                 |  |                           |
| 2019/2020 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>- kolokwium</li> <li>- egzamin pisemny (dłuższa wypowiedź pisemna / rozwiązanie problemu)</li> </ul> |                           |
|   |                 | <b>Podstawowe kryteria oceny</b>   |                           |
|   |                 | Przedmiot kończy się egzaminem z pytaniami otwartymi.<br>Warunkiem przystąpienia do egzaminu jest uzyskanie pozytywnych ocen zarówno z ćwiczeń laboratoryjnych jak i ćwiczeń audytoryjnych.  |                           |
| <b>Sposób weryfikacji założonych efektów kształcenia</b>  |                 |  |                           |
| <b>Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi</b>   |                 |  |                           |
| <b>A. Wymagania formalne</b>  |                 |  |                           |
| brak wymagań formalnych   |                 |  |                           |

|   |   |
|---|---|
| <b>B. Wymagania wstępne</b><br>brak wymagań wstępnych   |   |
| <b>Cele kształcenia</b><br>Celem przedmiotu jest poznanie podstawowych technik programistycznych oraz zdobycie umiejętności projektowania, analizy i implementacji podstawowych algorytmów.   |   |
| <b>Treści programowe</b> <ul style="list-style-type: none"> <li>• Algorytm a program. Proste programy w różnych sposobach zapisu (opis słowny, schemat blokowy, instrukcje języka programowania). Ręczna symulacja działania algorytmu.</li> <li>• Schemat przetwarzania programu w języku wysokiego poziomu - etapy kompilacji, konsolidacji i wykonania.</li> <li>• Deklaracje zmiennych. Proste typy danych oraz typy strukturalne: tablica, rekord. Zakres deklaracji i widoczności zmiennych, zmienne globalne.</li> <li>• Instrukcje sterujące. Pojęcie poprawności częściowej i całkowitej programu. Dowodzenie poprawności metodą niezmienników. Zagnieżdżone pętle.</li> <li>• Procedury i funkcje. Sposoby przekazywania parametrów do funkcji.</li> <li>• Mechanizm rekursji i jego wykorzystanie. Wyrażanie pętli przez rekursję i przykład wyrażenia rekursji pętlą. Dowodzenie poprawności funkcji rekurencyjnych za pomocą indukcji matematycznej.</li> <li>• Szacowanie liczby operacji wykonywanych przez algorytm.</li> <li>• Reprezentacja liczb całkowitych i zmiennoprzecinkowych w komputerze oraz błędy nimi spowodowane (przekroczenie zakresu i błędy zaokrąglenia).</li> <li>• Wykorzystanie wskaźników i zarządzanie pamięcią na przykładzie prostych struktur danych.</li> <li>• Informacja o teoretycznych ograniczeniach obliczeniowych.</li> </ul> |   |
| <b>Wykaz literatury</b> <p>[1] Griffiths David, Griffiths Dawn. <i>C. Rusz głową!</i> Wydawnictwo Helion.</p> <p>[2] Szepietowski A. <i>Podstawy informatyki.</i> Wydawnictwo Uniwersytetu Gdańskiego, Gdańsk 2000.</p> <p>[3] Kernighan B.W., Ritchie D.M. <i>Język ANSI C.</i> Wydawnictwa Naukowo-Techniczne, Warszawa 2003.</p>   |   |
| <b>Kierunkowe efekty kształcenia</b><br><br>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,<br>K_W03 zna podstawowe metody projektowania, analizowania i programowania algorytmów,<br>K_W04 zna podstawowe konstrukcje programistyczne oraz struktury danych,<br>K_U01 potrafi zastosować wiedzę matematyczną do formułowania, analizowania i rozwiązywania prostych zadań związanych z informatyką,<br>K_U03 potrafi pracować indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów,<br>K_U05 potrafi pisać, uruchamiać i testować programy w wybranym środowisku programistycznym<br>K_U06 projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy; wykorzystuje podstawowe techniki algorytmiczne i struktur danych<br>K_U08 posługuje się przyjętymi formatami reprezentacji różnego rodzaju danych stosownie do sytuacji<br>K_K02 potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania         | <b>Wiedza</b><br><br>Student:<br>- zna podstawowe narzędzia programistyczne<br>- zna elementy programów komputerowych: zmienne, instrukcje proste i sterujące, funkcje<br>- ma podstawy wiedzy o sposobach analizy poprawności i złożoności algorytmów<br>- zna typowe algorytmów obliczania wyszukiwania i porządkowania<br>- zna mechanizm rekursji<br>- zna sposoby reprezentacji liczb w komputerze<br>- zna podstawy technik programistycznych: dziel i rządź, DP<br>- zna podstawowe struktury danych: stos, kolejka  |
|   | <b>Umiejętności</b><br><br>Student:<br>- umie przeprowadzić ręczną symulację działania prostego programu z pętlami lub rekursją<br>- umie zaprojektować i zaimplementować program z pętlami (również zagnieżdżonymi) oraz funkcjami<br>- umie uzasadnić, że napisany program działa zgodnie z (nieformalnymi) wymaganiami<br>- potrafi ocenić liczbę operacji, wykonywanych przez program z pętlami<br>- umie zaprojektować proste funkcje rekurencyjne<br>- potrafi zapisywać i odczytywać liczby jako ciągu bitów, zarówno w przypadku stało- jak i zmiennie-pozycyjnym |
|   | <b>Kompetencje społeczne (postawy)</b><br><br>Student:<br>- potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania   |

## Kontakt

[jdybiz@inf.ug.edu.pl](mailto:jdybiz@inf.ug.edu.pl)