



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



| | | | | | | | | |
|--|------------------|---|---|----------------------|---|----------------------|----------------------|----------------------|
| Nazwa przedmiotu | | Kod ECTS | | | | | | |
| Programowanie | | 13.2.0158 | | | | | | |
| Nazwa jednostki prowadzącej przedmiot | | | | | | | | |
| Instytut Fizyki Doświadczalnej | | | | | | | | |
| Studia | | | | | | | | |
| wydział | kierunek | poziom | pierwszego stopnia | | | | | |
| Wydział Matematyki, Fizyki i Informatyki | Fizyka | forma | stacjonarne | | | | | |
| | | moduł | fizyka | | | | | |
| | | specjalnościowy | Podstawowa | | | | | |
| specjalizacja | | | | | | | | |
| Nazwisko osoby prowadzącej (osób prowadzących) | | | | | | | | |
| dr Janusz Młodzianowski; dr Sławomir Werbowy | | | | | | | | |
| Formy zajęć, sposób ich realizacji i przypisana im liczba godzin | | | | | Liczba punktów ECTS | | | |
| Formy zajęć | | | | | 4 Przedmiot w wymiarze 30h wykładu i 30h ćwiczeń w laboratorium komputerowym | | | |
| Wykład, Ćw. laboratoryjne | | | | | | | | |
| Sposób realizacji zajęć | | | | | | | | |
| zajęcia w sali dydaktycznej | | | | | | | | |
| Liczba godzin | | | | | | | | |
| Ćw. laboratoryjne: 30 godz., Wykład: 30 godz. | | | | | | | | |
| Cykl dydaktyczny | | | | | | | | |
| 2016/2017 letni | | | | | | | | |
| Status przedmiotu | | | Język wykładowy | | | | | |
| obowiązkowy | | | polski | | | | | |
| Metody dydaktyczne | | | Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne | | | | | |
| - praca własna - ćwiczenia w laboratorium komputerowym, praca własna | | | Sposób zaliczenia | | | | | |
| | | | - Zaliczenie na ocenę - Zaliczenie (zał) | | | | | |
| | | | Formy zaliczenia | | | | | |
| | | | kolokwium | | | | | |
| | | | Podstawowe kryteria oceny | | | | | |
| | | | Umiejętność samodzielnego, poprawnego napisania prostego programu w dowolnym kompilatorze języka C/C++, Umiejętność wyjaśnienia celowości zastosowania, w pisamym programie, konkretnych konstrukcji języka C/C++, Umiejętność posługiwania się, przynajmniej w podstawowym zakresie, wierszem poleceń systemu Windows, Obecność na zajęciach. | | | | | |
| Sposób weryfikacji założonych efektów kształcenia | | | | | | | | |
| zakładany efekt kształcenia | Kolokwium | Ocena pracy z komputerem i oprogramowaniem | mtd. dydakt 3 | mtd. dydakt 4 | mtd. dydakt 5 | mtd. dydakt 6 | mtd. dydakt 7 | mtd. dydakt 8 |
| | | | | | | | | |
| Wiedza | | | | | | | | |
| K_W12 | + | + | | | | | | |
| Umiejętności | | | | | | | | |
| K_U12 | + | + | | | | | | |

| | |
|--|---|
| <p>Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi</p> <p>A. Wymagania formalne B. Wymagania wstępne Znajomość matematyki w zakresie 1 semestru studiów.</p> | |
| <p>Cele kształcenia</p> <p>Celem zajęć jest przeprowadzenie podstawowego kursu programowania w języku C/C++.</p> | |
| <p>Treści programowe</p> <ol style="list-style-type: none"> Przechowywanie i kodowanie informacji w komputerze. Kodowanie ISO646, ISO8859-*, CP***, CP****, Unicode, U2, IEEE754-1985. Techniczne fazy tworzenia programu. Edycja, kompilacja, scalanie i diagnostyka programu. Projektowanie i programowanie strukturalne. Cykl życia projektu, podział na fazy i ich charakterystyka. Projektowanie metodą krokowego udoskonalania. Techniki wspomagające projektowanie. Schematy blokowe, diagramy strukturalne, Warnier-Orr'a, diagramy N-S. Elementy notacji BNF. Dokumentacja. Pojęcie i przykłady algorytmów. Historia rozwoju i specyfika języków C/C++. Zintegrowane pakiety: Borland, Microsoft, GNU, Preprocesor, Biblioteki standardowe. Struktura programu w języku C. Konwencje nazewnictwa, komentarze, słowa kluczowe dyrektywy. Deklaracje i podstawowe typy zmiennych. Definiowanie własnych typów. Konwersje typów. Zmienne wskaźnikowe. Pojęcie wskaźnika. Operator adresu i zawartości. Arytmetyka na wskaźnikach. Instrukcje wyboru i powtarzania. Omówienie i przykłady zastosowania. Funkcja jako podstawowa konstrukcja języka C/C++. Pojęcie funkcji i prototypu funkcji, wywołanie funkcji. Parametry formalne i aktualne. Przesyłanie argumentów przez wartość i adres. Zwracanie wyniku przez nazwę funkcji. Parametry wywołania funkcji main(). Operatory. Operatory arytmetyczne, logiczne, bitowe, relacyjne, przypisania, operator przecinkowy. Pojęcie lokalności i globalności. Zmienne lokalne, globalne, automatyczne, statyczne, rejestrowe, zewnętrzne. Przesyłanie zmiennych. Zmienne dynamiczne, tworzenie i usuwanie zmiennych dynamicznych. Komunikacja z urządzeniami wejścia/wyjścia, operacje plikowe. Elementy projektowania i programowania obiektowego. Pojęcie i przykłady obiektów. Enkapsulacja. Polimorfizm. Dziedziczenie. Omówienie metod projektowania obiektowego. Elementy języka C++. Przeciążanie funkcji i operatorów. Argumenty domniemane funkcji. Konstrukcje obiektowe klasy, struktury i unii. Funkcja i metoda. Konstruktor i destruktor. Szablony. Elementy programowania w systemie Windows. Okno, i pętla komunikatów. WinMain(), Win32 API. | |
| <p>Wykaz literatury</p> <p>B. W. Kernighan, D. M. Ritchie, Język ANSI C, WNT 1988 J. Bielecki, ANSI C++, Intersoftland 1997 N. Wirth, Wstęp do programowania systematycznego, WNT 1978 D. Harel, Rzecz o istocie informatyki. Algorytmika, WNT 1992 E. Yourdon, Współczesna analiza strukturalna, WNT 1996 J. Martin, J. J. Odell, Podstawy metod obiektowych, WNT 1997 C. Petzold, Programowanie Windows, Microsoft</p> | |
| <p>Efekty kształcenia (obszarowe i kierunkowe)</p> <p>K_W12 zna podstawy analizy numerycznej, zna na poziomie podstawowym co najmniej jeden pakiet do obliczeń symbolicznych, zna podstawowe pakiety oprogramowania użytkowego do prezentacji wyników i analizy danych; zna podstawy programowania i inżynierii oprogramowania</p> <p>K_U12 potrafi skompilować, uruchomić, testować i udokumentować napisany samodzielnie program komputerowy</p> | <p>Wiedza</p> <p>Student ma opanowane podstawowe konstrukcje języka C/C++. Zna typy zmiennych (char, int, float, tablica i struktura). Zna składnię instrukcji wyboru (if, if-else, switch), instrukcji iteracji (for, do, while) oraz operatory działań arytmetyczno logicznych. Rozumie czym jest funkcja i prototyp funkcji. Wie na czym polegają mechanizmy przesyłania parametrów do funkcji (lista formalna, przesyłanie przez wartość, adres, referencję). Zna pojęcie wskaźnika. Wie na czym polegają różnice między językami C i C++. Zna pojęcie obiektu.</p> <p>Zna pojęcie algorytmu programu i technikę projektowania metodą schematów blokowych. Wie co to jest złożoność obliczeniowa i poprawność całkowita. Rozumie na czym polega metodologia projektowania i programowania strukturalnego. Zna technikę dekompozycji funkcjonalnej i krokowego udoskonalania.</p> |
| | <p>Umiejętności</p> <p>Student potrafi sformułować za pomocą techniki schematów blokowych prosty algorytm numeryczny a na jego podstawie napisać program w języku C. Umie zaprojektować program składający się z kilku funkcji. Potrafi uzasadnić celowość użycia listy formalnej funkcji z przesłaniem argumentów przez wartość i adres. Potrafi zmodyfikować, poprzez zastosowanie innych rodzajów instrukcji wyboru i/lub iteracji, wybrany fragment programu. otrafi napisać program odczytujący z zapisujących dane numeryczne do i z pliku na dysku.</p> |

| | |
|-------------------|--|
| | |
| | Kompetencje społeczne (postawy) |
| Kontakt | |
| fizjm@univ.gda.pl | |