

SYLABUS (KARTA PRZEDMIOTU/MODUŁU)

Nazwa przedmiotu/modułu (zgodna z zatwierdzonym programem studiów na kierunku) Programowanie aplikacji komputerowych		Punkty ECTS 4	Numer katalogowy
Nazwa w j. angielskim Computer Applications Development			
Jednostka(i) realizująca(e) przedmiot/moduł (instytut/katedra) Instytut Inżynierii Biosystemów			
Kierownik przedmiotu/modułu dr inż. Radosław J. Kozłowski			
Kierunek studiów Informatyka stosowana	Poziom Studia I stopnia	Profil ogólnoakademicki	Semestr 3
Specjalność -	Specjalizacja inżynierska -		
RODZAJE ZAJĘĆ I ICH WYMIAR GODZINOWY (zajęcia zorganizowane i praca własna studenta)			
Forma studiów: stacjonarne		Forma studiów: niestacjonarne	
- wykłady	15	- wykłady	10
- ćwiczenia	20	- ćwiczenia	20
- inne z udziałem nauczyciela	10	- inne z udziałem nauczyciela	10
- praca własna	55	- praca własna	60
Łączna liczba godzin: 100		Łączna liczba godzin: 100	
CEL PRZEDMIOTU/MODUŁU			
<p>Celem przedmiotu jest uzyskanie przez studenta uniwersalnej wiedzy i umiejętności praktycznych z zakresu podstaw programowania aplikacji komputerowych w języku C, C++ i C#. W ramach zajęć omawiane i implementowane są elementarne zasady algorytmiki oraz elementy programowania takie jak: typy danych, stałe, zmienne, operacje wejścia i wyjścia, operatory, instrukcje, funkcje własne, tablice, wskaźniki, struktury oraz sposoby łączenia struktur z wskaźnikami i tablicami. Omówione zostają także metody i funkcje umożliwiające obsługę plików z danymi. Studenci zapoznają się ze strukturą typowego programu komputerowego w tym układem poszczególnych elementów składowych, zasięgiem zmiennych, przygotowaniem i uruchomieniem aplikacji, kompilacją warunkową, analizą i interpretacją napotkanych błędów. W ramach wykładów i ćwiczeń przeprowadzona jest analiza przykładowych programów o różnym stopniu złożoności.</p> <p>Po zakończonym kursie student nabywa wiedzę oraz zbiór praktycznych umiejętności pozwalających na samodzielne lub grupowe opracowanie algorytmów komputerowych oraz ich implementację w postaci programu komputerowego. Opracowane oprogramowanie pozwala na wspomaganie procesów z zakresu problemów inżynierskich.</p>			
METODY DYDAKTYCZNE			
<ol style="list-style-type: none"> 1. Ćwiczenia w sali komputerowej – samodzielna implementacja programów. 2. Wykłady – przedstawienie wiedzy teoretycznej. 3. Wykłady – prezentacja i analiza przykładowych programów. 			
EFEKTY KSZTAŁCENIA			Odniesienie do efektów kierunkowych
Wiedza	E1. Zna budowę oraz zasady tworzenia programów komputerowych w językach wysokiego poziomu. E2. Zna zasady budowy interfejsów tekstowych w programach komputerowych oraz sposoby komunikacji dwustronnej z użytkownikiem. E3. Ma wiedzę z zakresu instrukcji oraz wykorzystywanych w nich operatorów. E4. Zna zasady budowy oraz działania funkcji w aplikacji komputerowej. E5. Zna zasady definiowania i wykorzystania tablic jedno i wielowymiarowych. E6. Zna pojęcie wskaźników oraz możliwości ich wykorzystania w tablicach i funkcjach. E7. Zna możliwości struktur oraz zasady ich deklarowania i definiowania. E8. Zna metody komunikacji programu z tekstowymi, binarnymi i bazodanowymi plikami zewnętrznymi będącymi źródłem danych. E9. Zna metody testowania programów komputerowych.		IS1A_W17
Umiejętności	E10. Umie opracować algorytm i zaimplementować program komputerowy komunikujący się z użytkownikiem za pośrednictwem konsoli tekstowej. E11. Umie wytworzyć kod programu gromadzącego, przetwarzającego i analizującego duże zbiory danych zarówno numerycznych jak i tekstowych. E12. Umie wytworzyć kod programu, którego zadaniem jest rozwiązanie problemów algorytmicznych w tym problemów decyzyjnych oraz wymagających działań iteracyjnych i rekurencyjnych. E13. Umie kontrolować i zarządzać procesem wykorzystania przez wytworzone oprogramowanie pamięci operacyjnej i dyskowej komputera. E14. Umie analizować i interpretować informacje pozyskane na etapie walidacji i weryfikacji opracowanego kodu programu. E15. Ma umiejętność posługiwania się pakietem programistycznym Visual Studio w zakresie budowy aplikacji konsolowych.		IS1A_U13

Kompetencje społeczne	<p>E16. Ma świadomość możliwości metod programowania oraz zakresu ich zastosowania.</p> <p>E17. Wykazuje kreatywność w zakresie rozwiązywania problemów z wykorzystaniem programowania.</p> <p>E18. Ma umiejętności współpracy w zespole w celu rozwiązania wielokryterialnych problemów z wykorzystaniem programowania.</p> <p>E19. Ma świadomość o dynamicznie rozwijających się technikach programowania i konieczności ciągłego doskonalenia swoich umiejętności.</p>	IS1A_K05 IS1A_K06
<p>Metody weryfikacji efektów kształcenia</p> <ol style="list-style-type: none"> 1. Kontrola programów realizowanych w ramach ćwiczeń 2. Kolokwia przy komputerze sprawdzające umiejętności praktyczne 3. Egzamin ustny z pisemnym przygotowaniem. 		Numery efektów E1 – E19
<p style="text-align: center;">TREŚCI KSZTAŁCENIA</p> <ul style="list-style-type: none"> • Wprowadzenie do programowania. Przyjęcie języka C/C++/C# (bez elementów obiektowości) jako języka odniesienia. • Podstawowe elementy języka programowania: zestaw znaków, typy danych, stałe, zmienne, deklaracje, wyrażenia i instrukcje. • Struktura programu: zasięg nazw, ogólna struktura programu, przygotowanie i uruchomienie programu (notacja algorytmu, edycja tekstu programu, kompilacja programu, testowanie programu), kompilacja warunkowa, budowa programów wieloskładnikowych. • Operacje wyjścia i wejścia – wyprowadzanie i wprowadzanie wartości, formatowanie danych. • Operatory – arytmetyczne, porównania, logiczne, bitowe, przypisania, unarne, rozmiaru, konwersji, warunkowy, wskazywania. Priorytety operatorów. Konwersje jawne i niejawne. • Instrukcje – bloku, przypisania, warunkowa, cyklu, zaniechania, kontynuowania, wyboru, wywołania funkcji, powrotu, skoku, pusta. • Funkcje – definiowanie, wywołanie, rekurencja, funkcje biblioteczne, makrodefinicje. • Tablice – deklarowanie, inicjowanie, dostęp do elementów tablicy, powiązanie tablic z funkcjami. • Wskaźniki – wprowadzenie i zastosowania, używanie wskaźników z funkcjami i tablicami. • Struktury – wprowadzenie i zastosowania, używanie struktur z wskaźnikami i tablicami. • Łańcuchy – zmienne typu char, string; operacje na łańcuchach. • Pliki – wprowadzenie, otwieranie, zamykanie, zapis i odczyt. • Analiza przykładowych programów o wysokim stopniu złożoności. 		
<p>Formy i kryteria zaliczenia przedmiotu/modułu</p> <p>Kolokwium zaliczeniowe - w ramach przedmiotu przeprowadzi się jedno pisemne kolokwium. Zaliczenie wykładów (egzamin) - zaliczenie odbywa się w formie ustnej.</p>		<p>Procentowy udział w końcowej ocenie 100% oceny z ćwiczeń 100% oceny z wykładów</p>
<p style="text-align: center;">WYKAZ LITERATURY</p> <ol style="list-style-type: none"> 1. Ahno A.V., Ullman J.D. 2003. Wykłady z informatyki z przykładami w języku C. Helion. Gliwice. 2. Allain A. 2014. C++. Przewodnik dla początkujących. Helion. Gliwice. 3. Stroustrup B. 2014. Język C++. Kompendium wiedzy. Helion. Gliwice. 4. Eckel B. 2008 Thinking in C++. Edycja polska. Helion. Gliwice. 5. Majczak A. 2010. Programowanie strukturalne i obiektowe. Podręcznik do nauki zawodu technik informatyk. Helion. Gliwice. 6. Grebosz J. 2006. Symfonia C++ standard. Edition 2000. Kraków. 7. Prata S. 2012. Język C++. Szkoła programowania. Helion. Gliwice 8. Sokół R. 2005. Wstęp do programowania w języku C++. Helion. Gliwice. <p>i nowsze wydania...</p>		