

Nazwa modułu (blok przedmiotów): PROGRAMOWANIE		Kod modułu: M8													
Nazwa przedmiotu: Podstawy programowania		Kod przedmiotu:													
Wypełnia Zespół Kierunku								Nazwa jednostki prowadzącej przedmiot / moduł: INSTYTUT INFORMATYKI STOSOWANEJ							
								Nazwa kierunku: INFORMATYKA							
								Forma studiów: niestacjonarne		Profil kształcenia: PRAKTYCZNY			Specjalność:		
								Rok / semestr: 1/2		Status przedmiotu / modułu: obowiązkowy			Język przedmiotu / modułu: polski		
								Forma zajęć	wykład	ćwiczenia	ćwiczenia laboratoryjne	konwersatorium	seminarium	inne (wpisać jakie)	
								Wymiar zajęć	15		15				
								Koordynator przedmiotu / modułu		dr inż. Maciej Czyżak					
								Prowadzący zajęcia		dr inż. Maciej Czyżak, dr inż. Jerzy Skurczyński, dr Stefan Sokołowski, mgr inż. Mariusz Mazurek					
Cel przedmiotu / modułu		Ogólnym celem przedmiotu jest zapoznanie studenta z podstawami programowania strukturalnego z użyciem języka C dla standardów ANSI C i C99. Nauczenie tworzenia prostych aplikacji konsolowych w języku C. Przekazanie wiedzy w zakresie posługiwania się podstawowymi typami i konstrukcjami języka i funkcjami standardowymi. Nauczenie tworzenia własnych funkcji, stosowania prostych wskaźników i zmiennych dynamicznych, przetwarzania tekstów jak też zapisu informacji do plików i przetwarzania plików.													
Wymagania wstępne		Umiejętność posługiwania się systemem operacyjnym komputera, znajomość matematyki w zakresie programu szkoły średniej.													
EFEKTY KSZTAŁCENIA							Odniesienie do efektów dla programu								
Nr	Wiedza														
01	zna podstawowe typy w języku C						K_W07, K_W11								
02	zna podstawowe konstrukcje gramatyczne języka C						K_W07, K_W11								
03	rozumie pojęcie funkcji, zna budowę funkcji i metody komunikacji funkcji z otoczeniem, zna podstawowe funkcje standardowe						K_W07, K_W11								
04	ma podstawową wiedzę w zakresie wskaźników, zmiennych dynamicznych, tablic znakowych, struktur i plików						K_W07, K_W11								
Umiejętności															
05	pisze proste aplikacje konsolowe w języku C realizujące operacje wejścia/wyjścia, stosujące typy języka C, wyrażenia z użyciem operatorów arytmetycznych i logicznych, instrukcje warunkowe, pętle, tablice i funkcje standardowe, tworzy i wykorzystuje własne funkcje						K_U01, K_U07 K_U16, K_U17								
06	stosuje proste wskaźniki i zmienne dynamiczne						K_U01 K_U16 K_U17								
07	wyprowadza informacje do plików i wykonuje podstawowe czynności związane z przetwarzaniem plików						K_U01 K_U16 K_U17								
Kompetencje społeczne															
08	rozumie znaczenie pracy własnej, umie też współpracować w ramach małych zespołów, potrafi wyszukiwać informacje związane z językami programowania						K_K01								
TREŚCI PROGRAMOWE															
Forma zajęć – WYKŁAD															
Ogólna charakterystyka języka ANSI C. Przykłady prostych programów. Definicja zmiennych. Operator przypisania. Typy całkowite i operatory arytmetyki całkowitej. Operacje wejścia/wyjścia dla typów całkowitych. Operatory relacyjne i logiczne. Identyfikatory. Słowa kluczowe. Typy znakowe. Operacje wejścia/wyjścia dla typów rzeczywistych. Standardowe funkcje arytmetyczne. Konwersje arytmetyczne. Instrukcja if. Instrukcja if else. Zagnieżdżona instrukcja if. Operator warunkowy.															

Operator przecinkowy. Instrukcja złożona. Tablice jednowymiarowe. Instrukcja while. Instrukcja do while. Instrukcja for. Instrukcja break i continue. Instrukcja switch. Instrukcja goto. Operator rzutowania. Operatory inkrementacji i dekrementacji. Priorytety operatorów. Funkcje cz. I. Lista parametrów. Przekazywanie przez wartość. Wskaźniki. Funkcje cz. II. Deklaracja funkcji. Przekazywanie przez zmienną. Tablice jako argumenty funkcji. Instrukcja return. tablice wielowymiarowe. Dynamiczna alokacja pamięci. Stałe symboliczne. Pliki nagłówkowe. makra. Program wieloplikowy. Klasy pamięci w programie jednoplikowym i wieloplikowym. Kwalifikatory typu. Biblioteka standardowa ANSI C. Tablice znaków. Wejście i wyjście dla tablic znaków. Funkcje łańcuchowe. Tablice i wskaźniki. Struktury. Struktury zagnieżdżone. Tablice struktur. Wskaźniki do struktur. Sortowanie. Pliki. Tryby otwarcia plików. Pliki tekstowe. Pliki binarne. Podstawowe operacje plikowe. Konstrukcja typedef. Typy wyliczeniowe. Wskaźniki do funkcji. Argumenty linii wywołania. Unia i pola bitowe.

Forma zajęć – LABORATORIUM

Celem laboratorium jest nauczenie studentów projektowania, edycji, kompilacji i diagnostyki prostych programów konsolowych języku C. W ramach laboratorium uruchamiane są programy ilustrujące poszczególne konstrukcje języka C, następnie student tworzy własne programy z użyciem elementów gramatyki języka C prezentowanych na wykładzie.

1. Wprowadzenie do środowiska Dev C++, tworzenie projektu, edycja kodu źródłowego, kompilacja projektu, usuwanie błędów i uruchamianie. Definicja zmiennych typu całkowitego. Realizacja wprowadzania i wyprowadzania danych dla typów całkowitych, obliczanie prostych wyrażeń.
2. Definiowanie zmiennych dla typów rzeczywistych, wprowadzanie i wyprowadzanie danych dla typów rzeczywistych. Obliczanie wyrażeń z użyciem typów rzeczywistych. Podstawowe formatowanie wydruku. Kontrola poprawności danych wejściowych. Stosowanie operatorów relacyjnych i logicznych. Zapis funkcji logicznych. Obliczanie wyrażeń z użyciem standardowych funkcji arytmetycznych.
3. Zastosowanie instrukcji warunkowych i instrukcji przełącznikowej. Stosowanie instrukcji iteracyjnych, operatorów inkrementacji i dekrementacji. Realizacja przykładowych działań na tablicach jedno- i dwuwymiarowych. Sortowanie tablic.
4. Definiowanie i stosowanie funkcji. Proste operacje na wskaźnikach i użycie wskaźników do działań na tablicach. Implementacja komunikacji między funkcjami. Tworzenie zmiennych i tablic dynamicznych i działania na obiektach dynamicznych.
5. Wczytywanie i drukowanie stałych i zmiennych znakowych oraz tablic znakowych. Stosowanie podstawowych funkcji łańcuchowych. Definiowanie zmiennych i tablic typów strukturalnych i działania na obiektach typów strukturalnych. Tworzenie plików, zapis informacji do plików i odczyt, realizacja prostego przetwarzania plików.

Metody kształcenia	Wykład z prezentacją wprowadzający podstawowe informacje o języku C dla standardów ANSI C i C99 i wykorzystaniu ich do tworzenia programów. Ćwiczenia laboratoryjne - studenci uruchamiają programy ilustrujące konstrukcje zaprezentowane na wykładzie, następnie tworzą i uruchamiają programy, wykorzystujące przedstawione elementy języka, ale o większym stopniu złożoności. Po każdym ćwiczeniu laboratoryjnym student przygotowuje sprawozdanie zawierające opracowane i uruchomione programy wraz z wynikami ich działania. Student po serii ćwiczeń laboratoryjnych otrzymuje do wykonania zestaw zadań domowych.
Metody weryfikacji efektów kształcenia	
Nr efektu kształcenia z sylabusu	
Kolokwia w trakcie semestru, weryfikacja sprawozdań z ćwiczeń laboratoryjnych	05,06,07,08
Egzamin	01,02,03,04
Forma i warunki zaliczenia	Egzamin pisemny na koniec semestru. 3 kolokwia w ciągu semestru na laboratorium, dostarczenie sprawozdań i prac domowych –
Literatura podstawowa	<ol style="list-style-type: none"> 1. A.R. Neibauer, Języki C i C++, Twój pierwszy program, Help 1995, 2000 2. G. Perry, Język C w przykładach, Que, W-wa, 2000 3. B. Kernighan, D. Ritchie, Język ANSI C, WNT, 2000
Literatura uzupełniająca	
NAKLAD PRACY STUDENTA:	
	Liczba godzin
Udział w wykładach	15
Samodzielne studiowanie tematyki wykładów	20
Udział w ćwiczeniach audytoryjnych i laboratoryjnych*	15
Samodzielne przygotowywanie się do ćwiczeń*	60

Przygotowanie projektu / eseju / itp. *	15
Przygotowanie się do egzaminu / zaliczenia	20
Udział w konsultacjach	5
Inne - egzamin	3
ŁĄCZNY nakład pracy studenta w godz.	153
Liczba punktów ECTS za przedmiot	6
Obciążenie studenta związane z zajęciami praktycznymi*	90 3,6 ECTS
Obciążenie studenta na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich	38 1,6 ECTS