

**AGH**AGH UNIVERSITY OF SCIENCE  
AND TECHNOLOGY

Nazwa modułu: Programowanie proceduralne

Rok akademicki: 2015/2016      Kod: BIT-1-304-s      Punkty ECTS: 5

Wydział: Geologii, Geofizyki i Ochrony Środowiska

Kierunek: Informatyka Stosowana      Specjalność: —

Poziom studiów: Studia I stopnia      Forma i tryb studiów: Stacjonarne

Język wykładowy: Polski      Profil kształcenia: Ogólnoakademicki (A)      Semestr: 3

Strona www: —

Osoba odpowiedzialna: dr inż. Bała Justyna (jbala@geol.agh.edu.pl)

Osoby prowadzące: dr Onderka Zdzisław (zonderka@agh.edu.pl)

## Opis efektów kształcenia dla modułu zajęć

Kod EKM	Student, który zaliczył moduł zajęć wie/umie/potrafi	Powiązania z EKK	Sposób weryfikacji efektów kształcenia (forma zaliczeń)
Wiedza			
M_W001	Student ma uporządkowaną wiedzę w zakresie metodyki i technik programowania	IT1A_W11	Kolokwium
Umiejętności			
M_U001	Student potrafi zaprojektować oraz zrealizować prostą aplikację informatyczną, używając właściwych metod, technik i narzędzi	IT1A_U13	Kolokwium, Wykonanie projektu
M_U002	Student potrafi łączyć znane algorytmy obliczeniowe i tworzyć własne w celu rozwiązywania prostych zagadnień obliczeniowych	IT1A_U15	Kolokwium, Wykonanie projektu
M_U003	Student potrafi zastosować zasady rozumowania algorytmicznego do rozwiązywania problemów	IT1A_U16	Projekt, Udział w dyskusji
Kompetencje społeczne			
M_K001	Student posiada umiejętność współpracy i posiada zdolność do samokształcenia	IT1A_K01, IT1A_K03	Kolokwium, Projekt

## Matryca efektów kształcenia w odniesieniu do form zajęć

Kod EKM	Student, który zaliczył moduł zajęć wie/umie/potrafi	Forma zajęć										
		Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Inne	E-learning
Wiedza												
M_W001	Student ma uporządkowaną wiedzę w zakresie metodyki i technik programowania	+	-	+	-	-	-	-	-	-	-	-
Umiejętności												
M_U001	Student potrafi zaprojektować oraz zrealizować prostą aplikację informatyczną, używając właściwych metod, technik i narzędzi	-	-	+	-	-	-	-	-	-	-	-
M_U002	Student potrafi łączyć znane algorytmy obliczeniowe i tworzyć własne w celu rozwiązywania prostych zagadnień obliczeniowych	-	-	+	-	-	-	-	-	-	-	-
M_U003	Student potrafi zastosować zasady rozumowania algorytmicznego do rozwiązywania problemów	-	-	+	-	-	-	-	-	-	-	-
Kompetencje społeczne												
M_K001	Student posiada umiejętność współpracy i posiada zdolność do samokształcenia	-	-	+	-	-	-	-	-	-	-	-

## Treść modułu zajęć (program wykładów i pozostałych zajęć)

### Wykład

1. Przypomnienie podstawowych elementów języka C (typy danych, instrukcje proste i strukturalne, operacje we/wy standardowe i strumieniowe).
2. Przypomnienie problematyki tablic jednowymiarowych i dwuwymiarowych. Generowanie liczb pseudolosowych. Operacje na tablicach jednowymiarowych i dwuwymiarowych
3. Tablice wielowymiarowe
4. Wskaźniki, adresy pamięci. Typ wskaźnikowy, wskaźniki zmiennych, arytmetyka na adresach, wskaźnikach. Praca z tablicami w zapisie indeksowym i wskaźnikowym. Tablica a wskaźnik
5. Zarządzanie pamięcią. Alokacja i zwalnianie pamięci przydzielonej dynamicznie.
6. Tworzenie własnych funkcji. Sposoby przekazywania parametrów, przekazywanie tablic do funkcji.
7. Operacje na znakach i tablicach znaków.
8. Operacje na plikach tekstowych i binarnych.
9. Struktury, tablice struktur, wskaźniki do struktur, zapis struktur do pliku.
10. Dynamiczne struktury: listy jedno- i dwukierunkowe, drzewa binarne.

- 11.Optymalizacja kodu.
- 12.Profilowanie funkcji i pamięci.

### **Ćwiczenia laboratoryjne**

- 1.Przypomnienie podstawowych elementów języka C (typy danych, instrukcje proste i strukturalne, operacje we/wy standardowe i strumieniowe).
- 2.Przypomnienie problematyki tablic jednowymiarowych i dwuwymiarowych. Generowanie liczb pseudolosowych. Operacje na tablicach jednowymiarowych i dwuwymiarowych
- 3.Tablice wielowymiarowe
- 4.Wskaźniki, adresy pamięci. Typ wskaźnikowy, wskaźniki zmiennych, arytmetyka na adresach, wskaźnikach. Praca z tablicami w zapisie indeksowym i wskaźnikowym. Tablica a wskaźnik
- 5.Zarządzanie pamięcią. Alokacja i zwalnianie pamięci przydzielonej dynamicznie.
- 6.Tworzenie własnych funkcji. Sposoby przekazywania parametrów, przekazywanie tablic do funkcji.
- 7.Operacje na znakach i tablicach znaków.
- 8.Operacje na plikach tekstowych i binarnych.
- 9.Struktury, tablice struktur, wskaźniki do struktur, zapis struktur do pliku.
- 10.Dynamiczne struktury: listy jedno- i dwukierunkowe, drzewa binarne.
- 11.Optymalizacja kodu.
- 12.Profilowanie funkcji i pamięci.

### **Sposób obliczania oceny końcowej**

Ocena końcowa odpowiada ocenie z zaliczenia

### **Wymagania wstępne i dodatkowe**

Podstawowa wiedza informatyczna. Podstawy języka C  
Student powinien mieć zaliczony przedmiot: „Wstęp do informatyki”.

### **Zalecana literatura i pomoce naukowe**

- Brian W. Kernighan, Dennis M. Ritchie, Język ANSI C
- R. Jones, I. Stewart, Sztuka programowania w języku C
- C. Delannoy, Ćwiczenia z języka C
- R. Konieczny, Wprowadzenie do programowania w języku C
- J. Bielecki, Język C - interpretacja standardu
- N.Wirth, Algorytmy + Struktury Danych = Programy

### **Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu**

Nie podano dodatkowych publikacji

### **Informacje dodatkowe**

udział „praktycznych” punktów ECTS: 3,5

udział „teoretycznych” punktów ECTS: 1,5

Podstawowym terminem uzyskania zaliczenia jest koniec zajęć w danym semestrze. Student może przystąpić do poprawkowego zaliczenia dwukrotnie, w terminie wyznaczonym przez prowadzącego zajęcia.

Student który bez usprawiedliwienia opuścił więcej niż 20% zajęć może zostać pozbawiony przez prowadzącego możliwości poprawkowego zaliczania.

**Nakład pracy studenta (bilans punktów ECTS)**

Forma aktywności studenta	Obciążenie studenta
Udział w wykładach	14 godz
Samodzielne studiowanie tematyki zajęć	25 godz
Udział w ćwiczeniach laboratoryjnych	42 godz
Przygotowanie do zajęć	45 godz
Sumaryczne obciążenie pracą studenta	126 godz
Punkty ECTS za moduł	5 ECTS