

**AGH**AGH UNIVERSITY OF SCIENCE  
AND TECHNOLOGY

Nazwa modułu zajęć:	Techniki programowania				
Rok akademicki:	2019/2020	Kod:	CIMT-1-027-s	Punkty ECTS:	2
Wydział:	Inżynierii Materiałowej i Ceramiki				
Kierunek:	Inżynieria Materiałowa	Specjalność:	—		
Poziom studiów:	Studia I stopnia	Forma studiów:	Stacjonarne		
Język wykładowy:	Polski	Profil:	Ogólnoakademicki (A)	Semestr:	0
Strona www:	—				
Prowadzący moduł:	Szyszkievicz-Warzecha Krzysztof (szyszkin@agh.edu.pl)				

## Opis efektów uczenia się dla modułu zajęć

Kod MEU	Student, który zaliczył moduł zajęć zna i rozumie/potrafi/jest gotów do	Powiązania z KEU	Sposób weryfikacji i oceny efektów uczenia się osiągniętych przez studenta w ramach poszczególnych form zajęć i dla całego modułu zajęć
Wiedza: zna i rozumie			
M_W001	Student ma podstawową wiedzę z zakresu programowania w języku C++	IMT1A_W02	Aktywność na zajęciach, Kolokwium
M_W002	Student ma podstawową wiedzę z zakresu programowania obiektowego	IMT1A_W02	Aktywność na zajęciach, Kolokwium
Umiejętności: potrafi			
M_U001	Student potrafi napisać oraz skompilować prosty program w języku C	IMT1A_U02	Aktywność na zajęciach, Kolokwium
M_U002	Student potrafi wykorzystać wybrane środowisko programistyczne C	IMT1A_U02	Aktywność na zajęciach

**Liczba godzin zajęć w ramach poszczególnych form zajęć**

Suma	Forma zajęć dydaktycznych										
	Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Prace kontrolne i przejściowe	Lektorat
30	0	0	0	0	0	30	0	0	0	0	0

**Matryca kierunkowych efektów uczenia się w odniesieniu do form zajęć i sposobu zaliczenia, które pozwalają na ich uzyskanie**

Kod MEU	Student, który zaliczył moduł zajęć zna i rozumie/potrafi/jest gotów do	Forma zajęć dydaktycznych										
		Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Prace kontrolne i przejściowe	Lektorat
Wiedza: zna i rozumie												
M_W001	Student ma podstawową wiedzę z zakresu programowania w języku C++	-	-	-	-	-	+	-	-	-	-	-
M_W002	Student ma podstawową wiedzę z zakresu programowania obiektowego	-	-	-	-	-	+	-	-	-	-	-
Umiejętności: potrafi												
M_U001	Student potrafi napisać oraz skompilować prosty program w języku C	-	-	-	-	-	+	-	-	-	-	-
M_U002	Student potrafi wykorzystać wybrane środowisko programistyczne C	-	-	-	-	-	+	-	-	-	-	-

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

Forma aktywności studenta	Obciążenie studenta
Udział w zajęciach dydaktycznych/praktyka	30 godz
Przygotowanie do zajęć	10 godz
Samodzielne studiowanie tematyki zajęć	10 godz
Dodatkowe godziny kontaktowe	5 godz
Sumaryczne obciążenie pracą studenta	55 godz
Punkty ECTS za moduł	2 ECTS

## Pozostałe informacje

### Szczegółowe treści kształcenia w ramach poszczególnych form zajęć (szczegółowy program wykładów i pozostałych zajęć)

#### Zajęcia seminaryjne

##### Podstawowe pojęcia języka C++

Podstawy programowania w języku C++: struktura programu, pliki źródłowe, preprocesor, identyfikacja danych; narzędzia wspomagające programowanie: kompilator, linker, debugger; pliki nagłówkowe i zasady ich stosowania; najistotniejsze elementy języka C++;

##### Słowa kluczowe, typy danych

Podstawowe słowa kluczowe; typy danych; zmienne tablicowe, tablice jedno- i wielowymiarowe;

##### Operatory

operatory: arytmetyczne, relacyjne, logiczne, bitowe; instrukcje przypisania; instrukcje warunkowe proste i złożone; instrukcje wyboru; konstruowanie wyrażeń logicznych;

##### Instrukcje iteracyjne

instrukcje iteracyjne: while i do..while, instrukcja for; wskaźniki: arytmetyka wskaźnikowa, wskaźniki a tablice; struktury i unie;

##### Dynamiczne operacje na pamięci

dynamiczne przypisywanie i zwalnianie pamięci: operatory new i delete; funkcje: przekazywanie argumentów, referencje, przeciążanie funkcji, wskaźniki;

##### programowanie strukturalne

programowanie strukturalne – przykładowe programy (rozwiązanie drugiego prawa Ficka);

##### programowanie rekurencyjne

programowanie rekurencyjne – przykładowe programy (metody odwrotne dotyczące znajdowania współczynników dyfuzji).

##### Wprowadzenie do programowania obiektowego w języku C++

Wprowadzenie do programowania obiektowego w języku C++: podstawowe pojęcia: klasa, hermetyzacja, dziedziczenie, przeciążanie funkcji i operatorów, polimorfizm, obiekt;

##### Obiekty

projektowanie klas, strukturalizacja, hermetyzacja i komunikacja, metody, wewnętrzna struktura obiektów;

##### konstruktory i destruktory

konstruktory i destruktory obiektów;

##### komunikacja z obiektami

komunikacja z obiektami: funkcje składowe i zaprzyjaźnione, zmienna this;

##### dziedziczenie

dziedziczenie: konstruowanie obiektów klasy pochodnej, funkcje wirtualne, klasy abstrakcyjne i fundamentalne;

##### Programowanie obiektowe a strukturalne

porównanie programowania strukturalnego z obiektowym, najważniejsze różnice.

### **Metody i techniki kształcenia:**

Zajęcia seminaryjne: Na zajęciach seminaryjnych podstawą jest prezentacja multimedialna oraz ustna prowadzona przez studentów. Kolejnym ważnym elementem kształcenia są odpowiedzi na powstałe pytania, a także dyskusja studentów nad prezentowanymi treściami.

### **Warunki i sposób zaliczenia poszczególnych form zajęć, w tym zasady zaliczeń poprawkowych, a także warunki dopuszczenia do egzaminu:**

Nie określono

### **Zasady udziału w poszczególnych zajęciach, ze wskazaniem, czy obecność studenta na zajęciach jest obowiązkowa:**

Zajęcia seminaryjne:

- Obecność obowiązkowa: Tak
- Zasady udziału w zajęciach: Studenci prezentują na forum grupy temat wskazany przez prowadzącego oraz uczestniczą w dyskusji nad tym tematem. Ocenie podlega zarówno wartość merytoryczna prezentacji, jak i tzw. kompetencje miękkie.

### **Sposób obliczania oceny końcowej**

Średnia ocen z kolokwiów

### **Sposób i tryb wyrównywania zaległości powstałych wskutek nieobecności studenta na zajęciach:**

Nie określono

### **Wymagania wstępne i dodatkowe, z uwzględnieniem sekwencyjności modułów**

Wymagania wstępne: Informatyka

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

1. J. Grębosz, Symfonia C++, Oficyna Kallimach, 1996 lub nowsze.
2. Stephen Prata, Język C++. Szkoła programowania. Helion 2006.
3. Ogólnodostępne podręczniki do języka C++.

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

Nie podano dodatkowych publikacji

### **Informacje dodatkowe**

Brak