

**AGH**AGH UNIVERSITY OF SCIENCE
AND TECHNOLOGY

Nazwa modułu zajęć:	Techniki informatyczne				
Rok akademicki:	2019/2020	Kod:	RIMM-1-105-s	Punkty ECTS:	4
Wydział:	Inżynierii Mechanicznej i Robotyki				
Kierunek:	Inżynieria Mechaniczna i Materiałowa	Specjalność:	—		
Poziom studiów:	Studia I stopnia	Forma studiów:	Stacjonarne		
Język wykładowy:	Polski	Profil:	Ogólnoakademicki (A)	Semestr:	1
Strona www:	—				
Prowadzący moduł:	dr hab. inż. Banaś Marian (mbanas@agh.edu.pl)				

Treści programowe zapewniające uzyskanie efektów uczenia się dla modułu zajęć

Wprowadzenie studenta w problematykę budowy i użytkowania współczesnych systemów informatycznych, zapoznanie z podstawami programowania w językach C oraz C++.

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	Zna rolę systemów operacyjnych. Zna współczesne systemy operacyjne. Zna charakterystykę systemów Posix na przykładzie Unix/Linux. Zna podstawowe zasady funkcjonowania systemów Unix. Wie jak zarządzać zasobami konta, procesami.	IMM1A_W04, IMM1A_W12	Aktywność na zajęciach, Kolokwium
M_W002	Zna budowę, modele topologie i zasady funkcjonowania sieci komputerowych. Wie co to jest Internet i zna jego podstawowe usługi. Ma świadomość zagrożeń bezpieczeństwa danych.	IMM1A_W04, IMM1A_W12	Kolokwium, Aktywność na zajęciach

M_W003	Posiada wiedzę o sposobach tworzenia oprogramowania. Zna rolę i zasady tworzenia algorytmów. Potrafi scharakteryzować kompilatory, interpretery. Zna paradygmaty programowania strukturalnego, obiektowego i funkcyjnego.	IMM1A_W04, IMM1A_W12	Aktywność na zajęciach, Kolokwium
M_W004	Zna elementy składowe języków programowania ANSI C i C++. Zna ich składnię i semantykę.	IMM1A_W04, IMM1A_W12	Aktywność na zajęciach, Kolokwium
Umiejętności: potrafi			
M_U001	Umie się posługiwać systemami wielodostępnymi na przykładzie Linuxa. Umie zarządzać procesami, rozpoznać otoczenie, dobrać właściwe oprogramowanie terminalowe do pracy z systemami unixowymi.	IMM1A_U10, IMM1A_U07, IMM1A_U01, IMM1A_U09	Udział w dyskusji, Wykonanie ćwiczeń laboratoryjnych
M_U002	Umie stworzyć, skompilować i przeprowadzić debugowanie prostego programu w języku C. Umie zastosować funkcje biblioteczne (standardowe i rozszerzające) we własnych programach.	IMM1A_U10, IMM1A_U07, IMM1A_U01, IMM1A_U09	Wykonanie ćwiczeń laboratoryjnych, Udział w dyskusji
M_U003	Umie definiować struktury danych, pobierać, przetwarzać i prezentować dane użytkownika.	IMM1A_U10, IMM1A_U07, IMM1A_U01, IMM1A_U09	Wykonanie ćwiczeń laboratoryjnych, Udział w dyskusji
M_U004	Umie napisać program z użyciem języka C++. Umie zbudować klasę, potrafi posłużyć się mechanizmami dziedziczenia i polimorfizmu. Umie zdefiniować operatory użytkownika i posłużyć się szablonami STL.	IMM1A_U10, IMM1A_U07, IMM1A_U01, IMM1A_U09	Wykonanie ćwiczeń laboratoryjnych, Projekt
M_U005	Umie dobrać właściwy język programowania stosownie do wymagań systemowych i użytkownika.	IMM1A_U10, IMM1A_U07, IMM1A_U01, IMM1A_U09	Wykonanie ćwiczeń laboratoryjnych, Udział w dyskusji
Kompetencje społeczne: jest gotów do			
M_K001	Potrafi rozpoznać i ocenić przydatność narzędzi informatycznych spośród współcześnie dostępnych.	IMM1A_K04, IMM1A_K06, IMM1A_K01	Zaangażowanie w pracę zespołu, Udział w dyskusji, 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
56	28	0	28	0	0	0	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	Zna rolę systemów operacyjnych. Zna współczesne systemy operacyjne. Zna charakterystykę systemów Posix na przykładzie Unix/Linux. Zna podstawowe zasady funkcjonowania systemów Unix. Wie jak zarządzać zasobami konta, procesami.	+	-	-	-	-	-	-	-	-	-	-
M_W002	Zna budowę, modele topologie i zasady funkcjonowania sieci komputerowych. Wie co to jest Internet i zna jego podstawowe usługi. Ma świadomość zagrożeń bezpieczeństwa danych.	+	-	-	-	-	-	-	-	-	-	-
M_W003	Posiada wiedzę o sposobach tworzenia oprogramowania. Zna rolę i zasady tworzenia algorytmów. Potrafi scharakteryzować kompilatory, interpretery. Zna paradygmaty programowania strukturalnego, obiektowego i funkcyjnego.	+	-	-	-	-	-	-	-	-	-	-
M_W004	Zna elementy składowe języków programowania ANSI C i C++. Zna ich składnię i semantykę.	+	-	-	-	-	-	-	-	-	-	-
Umiejętności: potrafi												

M_U001	Umie się posługiwać się systemami wielodostępnymi na przykładzie Linuxa. Umie zarządzać procesami, rozpoznać otoczenie, dobrać właściwe oprogramowanie terminalowe do pracy z systemami unixowymi.	-	-	+	-	-	-	-	-	-	-	-
M_U002	Umie stworzyć, skompilować i przeprowadzić debugowanie prostego programu w języku C. Umie zastosować funkcje biblioteczne (standardowe i rozszerzające) we własnych programach.	-	-	+	-	-	-	-	-	-	-	-
M_U003	Umie definiować struktury danych, pobierać, przetwarzać i prezentować dane użytkownika.	-	-	+	-	-	-	-	-	-	-	-
M_U004	Umie napisać program z użyciem języka C++. Umie zbudować klasę, potrafi posłużyć się mechanizmami dziedziczenia i polimorfizmu. Umie zdefiniować operatory użytkownika i posłużyć się szablonami STL.	-	-	+	-	-	-	-	-	-	-	-
M_U005	Umie dobrać właściwy język programowania stosownie do wymagań systemowych i użytkownika.	-	-	+	-	-	-	-	-	-	-	-
Kompetencje społeczne: jest gotów do												
M_K001	Potrafi rozpoznać i ocenić przydatność narzędzi informatycznych spośród współcześnie dostępnych.	+	-	+	-	-	-	-	-	-	-	-

Nakład pracy studenta (bilans punktów ECTS)

Forma aktywności studenta	Obciążenie studenta
Udział w zajęciach dydaktycznych/praktyka	56 godz
Przygotowanie do zajęć	10 godz
przygotowanie projektu, prezentacji, pracy pisemnej, sprawozdania	30 godz
Samodzielne studiowanie tematyki zajęć	10 godz
Egzamin lub kolokwium zaliczeniowe	2 godz
Dodatkowe godziny kontaktowe	5 godz
Sumaryczne obciążenie pracą studenta	113 godz
Punkty ECTS za moduł	4 ECTS

Pozostałe informacje

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

Wykład

Systemy operacyjne. Systemy Posix. System Linux. Dystrybucje. Konta. Praca zdalna. Podstawowe polecenia systemu operacyjnego Unix/Linux. Biblioteki i narzędzia dla administratorów, programistów i użytkowników.

Protokoły i usługi sieciowe. Wykorzystanie i tworzenie zasobów internetu. Sieciowe systemy Operacyjne, Systemy czasu rzeczywistego. Zdalna administracja. Bezpieczeństwo sieci komputerowych.

Arytmetyka komputerów. Reprezentacja informacji w komputerze. Zapis binarny, ósemkowy, szesnastkowy. Kod ISO, ASCII, Unicode.

Podstawy programowania w języku C - składnia języka, typy danych, zmienne, instrukcje, operatory, funkcje, makrodefinicje, preprocesor, wejście/wyjście.

Struktury danych, komunikacja z użytkownikiem, projekt i implementacja algorytmu rozwiązania problemu informatycznego. Funkcje biblioteczne, struktury dynamiczne i algorytmy sortowania i optymalizacji.

Podstawy programowania w języku C++, Cechy programowania obiektowego. Strumienie. Przeładowanie funkcji, Adresy do funkcji. referencje.

Programowanie obiektowe. Klasy, dziedziczenie, polimorfizm. Mechanizm wyjątków.

Przeładowanie operatorów. Szablony funkcji i klas. Biblioteki STL.

Ćwiczenia laboratoryjne

Nabywanie praktycznych umiejętności z poniższych grup zagadnień:

Posługiwanie się wielodostępnymi sieciowymi systemami operacyjnymi
System operacyjny Linux. Wykorzystanie terminali dostępowych, konta i ich bezpieczeństwo, podstawowe polecenia systemu operacyjnego. Operacje na systemie plikowym, zarządzanie zasobami, archiwizacja. Wykorzystanie systemu wzorców nazw, strumieni i potoków.

Sieci komputerowych oraz usług sieci Internet

Poznanie budowy i działania sieci komputerowych. Rozpoznanie architektury sieci oraz adresowania i podstawowych usług TCP/IP. Wykorzystanie oprogramowania narzędziowego do wykorzystania i zarządzania administracją standardowych usług sieciowych (www, ftp, scp, e-mail). Tworzenie i konserwacja własnej strony WWW.

Programowanie strukturalne w języku C

Budowa programu. Wyprowadzanie sformatowanych tekstów. Dane, wyrażenia, stałe, zmienne, typy.. Operatory i wyrażenia. Instrukcje warunkowe, pętle, instrukcje wielokrotnego wyboru. Funkcje, argumenty, przekazywanie argumentów. Tablice. Algorytmy wyszukiwania, sortowania. Argumenty wywołania programu. Losowanie. Struktury i unie. Tworzenie bibliotek.

Programowanie obiektowe w języku C++

Nowe typy danych w C++. Strumienie wejścia/wyjścia. Wskaźniki i referencje.

Wskaźniki do funkcji. Tablice wielowymiarowe. Klasy. Dziedziczenie. Polimorfizm. Przeładowanie operatorów. Tworzenie szablonów funkcji oraz klas. Wykorzystanie składników biblioteki STL. Implementacja mechanizmów obsługi wyjątków. Tworzenie programów do rozwiązywania prostych problemów matematycznych i fizycznych, do analizy tekstów, do prostych obliczeń z wykorzystaniem bibliotek użytkowych.

Metody i techniki kształcenia:

Wykład: Treści prezentowane na wykładzie są przekazywane z użyciem komputera - omawiane zagadnienia są ilustrowane z użyciem środowiska IDE poprzez praktyczne przykłady odnoszące się do prezentowanych zagadnień. Przykłady te powstają również z interakcji prowadzącego ze studentami podczas wykładu.

Ćwiczenia laboratoryjne: Podczas ćwiczeń laboratoryjnych studenci rozwiązują indywidualnie otrzymane od prowadzącego zadania i są na bieżąco konsultowani przez prowadzącego zajęcia tak, aby otrzymane wyniki miały wysoką wartość merytoryczną.

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

Zaliczenie wszystkich zadań z ćwiczeń laboratoryjnych.

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

Wykład:

- Obecność obowiązkowa: Nie

- Zasady udziału w zajęciach: Studenci uczestniczą w zajęciach poznając kolejne treści nauczania zgodnie z sylabusem przedmiotu. Studenci winni na bieżąco zadawać pytania i wyjaśniać wątpliwości. Rejestracja audiowizualna wykładu wymaga zgody prowadzącego.

Ćwiczenia laboratoryjne:

- Obecność obowiązkowa: Tak

- Zasady udziału w zajęciach: Zaliczenie zajęć odbywa się na podstawie zaprezentowania rozwiązania postawionego problemu. Zaliczenie modułu jest możliwe po zaliczeniu wszystkich zajęć laboratoryjnych.

Sposób obliczania oceny końcowej

Średnia z zaliczenia laboratoriów i kolokwium.

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

Rozwiązanie zadań z ćwiczeń laboratoryjnych, na których student był nieobecny na następnych zajęciach lub w innym terminie w porozumieniu z prowadzącym zajęcia.

Wymagania wstępne i dodatkowe, z uwzględnieniem sekwencyjności modułów

Podstawowa znajomość obsługi komputera.

Zalecana literatura i pomoce naukowe

Sikorski W. : Podstawy Technik Informatycznych. PWN. Warszawa 2006.

Wojciechowski A. : Usługi w sieciach informatycznych. Wyd. MIKOM. Warszawa 2006.

Kernighan B.W. , Ritchie D.M. : Język ANSI C. WNT, Warszawa, 2004.

Grębosz J. : Symfonia C++. Wyd Edition. Kraków 2006.

Grębosz J. : Opus Magnum C++. Wyd Edition. Kraków 2018.

Banaś M.: Techniki Informatyczne. Przewodnik do ćwiczeń, Kraków 2016 i późn. (cykliczne materiały dla studentów WIMiR)

Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu

Banaś M.: Techniki Informatyczne. Przewodnik do ćwiczeń, Kraków 2016 i późn. (cykliczne materiały dla studentów WIMiR).

Banaś M., Marczakowski P.: Oprogramowanie komputerowe do projektowania zbiorników niskociśnieniowych na podstawie normy API 620. Studencka Sesja Naukowa, AGH, Kraków 1997

Banaś M., Migdalski J.: Automatyzacja procesu badań potencjometrycznych z użyciem wielokanałowego, skomputeryzowanego zestawu pomiarowego. Zeszyty Naukowe AGH, s. Mechanika, Kraków 2000, tom 19, 271-284.

Banaś M. Obtaining parameters of granulometric characteristics of suspension with usage computer controlled sedimentation balance. Visnik Nacional'nogo univrsitetu "L'vivs`ka politehnika" . ISSN 0321-0499. Komp'uterni sistemi proektuvannâ. 2004 no. 501, s. 62-68

Banaś M. Computer simulations of the sedimentation process. Vidavnictvo Nacional'nogo univrsitetu "L'vivs'ka politehnika". 2004. Pp. 244-247. Lviv.

Banaś M. Theoretical analysis and investigations of properties on non-grain suspensions used in design and exploitation of lamella sedimentation tanks. AGH. Kraków 2013.

Informacje dodatkowe

Możliwość wcześniejszego zaliczenia laboratoriów na podstawie wyniku realizacji indywidualnego zadania oraz przedstawieniu własnych osiągnięć.