

**AGH**AGH UNIVERSITY OF SCIENCE
AND TECHNOLOGY

Nazwa modułu zajęć: Programowanie obiektowe w języku C#

Rok akademicki: 2019/2020 Kod: RAIR-1-304-n Punkty ECTS: 3

Wydział: Inżynierii Mechanicznej i Robotyki

Kierunek: Automatyka i Robotyka Specjalność: —

Poziom studiów: Studia I stopnia Forma studiów: Niestacjonarne

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

Strona www: —

Prowadzący moduł: dr hab. inż. Banaś Marian (mbanas@agh.edu.pl)

Treści programowe zapewniające uzyskanie efektów uczenia się dla modułu zajęć

Zapoznanie studentów z programowaniem w językach zorientowanych obiektowo Java, C++ i C1. .

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 język programowania C++ oraz orientuje się w jego podstawowych implementacjach na przykładzie Visual C++ oraz CBuilder.	AIR1A_W10, AIR1A_W12	Aktywność na zajęciach, Udział w dyskusji
M_W002	Zna zasady tworzenia i korzystania z klas, dziedziczenia i polimorfizmu.	AIR1A_W10, AIR1A_W12	Aktywność na zajęciach, Udział w dyskusji
M_W003	Wie jak stworzyć własny program w języku C++, kodując w języku C++ rozwiązanie własnego problemu informatycznego.	AIR1A_W10, AIR1A_W12	Aktywność na zajęciach, Udział w dyskusji
M_W004	Zna podstawowe biblioteki C++ na wybranej platformie systemowej i wie jak rozszerzyć ich możliwości.	AIR1A_W10, AIR1A_W12	Aktywność na zajęciach, Udział w dyskusji
M_W005	Zna specyfikę języka Java. Wie co to jest B-Code. Wie w jakim środowisku uruchamiane są programy Jawy. Zna podstawowe rodzaje środowisk Jawy.	AIR1A_W10, AIR1A_W12	Aktywność na zajęciach, Udział w dyskusji

M_W006	Wie czym różnią się programy Jawy dedykowane dla wybranych środowisk (aplety, pełne programy graficzne).	AIR1A_W10, AIR1A_W12	Aktywność na zajęciach, Udział w dyskusji
M_W007	Zna zagadnienia i specyfikę programowania urządzeń telekomunikacyjnych oraz mobilnych.	AIR1A_W10, AIR1A_W12	Aktywność na zajęciach, Udział w dyskusji
M_W008	Ma świadomość i orientację, w jakim kierunku rozwijane są obiektowe języki (i systemy) programowania.	AIR1A_W10, AIR1A_W12	Aktywność na zajęciach, Udział w dyskusji
Umiejętności: potrafi			
M_U001	Potrafi wybrać właściwe dla siebie środowisko programowania obiektowego.	AIR1A_U04, AIR1A_U05	Projekt, Wykonanie ćwiczeń laboratoryjnych
M_U002	Potrafi w rozwiązywaniu własnego problemu informatycznego posłużyć się językiem C++: wykonać analizę problemu, zakodować, uruchomić i przeprowadzić proces debugowania programu.	AIR1A_U04, AIR1A_U05	Projekt, Wykonanie ćwiczeń laboratoryjnych
M_U003	Umie posłużyć się elementami oferowanymi przez system operacyjny oraz środowisko programowania dla uzyskania przyjaznego dla użytkownika interfejsu programu.	AIR1A_U04, AIR1A_U05	Projekt, Wykonanie ćwiczeń laboratoryjnych
M_U004	Umie wybrać właściwe dla siebie środowisko programistyczne/uruchomieniowe Jawy.	AIR1A_U04, AIR1A_U05	Projekt, Wykonanie ćwiczeń laboratoryjnych
M_U005	Umie tworzyć średnio zaawansowane aplety w Jawie.	AIR1A_U04, AIR1A_U05	Projekt, Wykonanie ćwiczeń laboratoryjnych
M_U006	Umie tworzyć proste aplikacje dla urządzeń mobilnych.	AIR1A_U04, AIR1A_U05	Projekt, Wykonanie ćwiczeń laboratoryjnych
M_U007	Umie wybrać właściwe dla specyficznych w danym miejscu pracy wymagań środowisko programowania obiektowego i potrafi się nim sprawnie posłużyć dla tworzenia i konserwacji oprogramowania użytkowego.	AIR1A_U04, AIR1A_U05	Projekt, Wykonanie ćwiczeń laboratoryjnych
Kompetencje społeczne: jest gotów do			
M_K001	Potrafi twórczo wykorzystywać aktualnie dostępne techniki tworzenia oprogramowania do własnych potrzeb w pracy zawodowej.	AIR1A_K03, AIR1A_K02, AIR1A_K01	Aktywność na zajęciach, Udział w dyskusji
M_K002	Potrafi poszukiwać i doskonalić się w zakresie nowych sposobów tworzenia przyjaznego użytkownikowi oprogramowania.	AIR1A_K03, AIR1A_K02, AIR1A_K01	Aktywność na zajęciach, Udział w dyskusji

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
20	12	0	8	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 język programowania C++ oraz orientuje się w jego podstawowych implementacjach na przykładzie Visual C++ oraz CBuilder.	+	-	-	-	-	-	-	-	-	-	-
M_W002	Zna zasady tworzenia i korzystania z klas, dziedziczenia i polimorfizmu.	+	-	-	-	-	-	-	-	-	-	-
M_W003	Wie jak stworzyć własny program w języku C++, kodując w języku C++ rozwiązanie własnego problemu informatycznego.	+	-	-	-	-	-	-	-	-	-	-
M_W004	Zna podstawowe biblioteki C++ na wybranej platformie systemowej i wie jak rozszerzyć ich możliwości.	+	-	-	-	-	-	-	-	-	-	-
M_W005	Zna specyfikę języka Java. Wie co to jest B-Code. Wie w jakim środowisku uruchamiane są programy Jawy. Zna podstawowe rodzaje środowisk Jawy.	+	-	-	-	-	-	-	-	-	-	-
M_W006	Wie czym różnią się programy Jawy dedykowane dla wybranych środowisk (aplety, pełne programy graficzne).	+	-	-	-	-	-	-	-	-	-	-
M_W007	Zna zagadnienia i specyfikę programowania urządzeń telekomunikacyjnych oraz mobilnych.	+	-	-	-	-	-	-	-	-	-	-

M_W008	Ma świadomość i orientację, w jakim kierunku rozwijane są obiektowe języki (i systemy) programowania.	+	-	-	-	-	-	-	-	-	-	-
Umiejętności: potrafi												
M_U001	Potrafi wybrać właściwe dla siebie środowisko programowania obiektowego.	-	-	+	-	-	-	-	-	-	-	-
M_U002	Potrafi w rozwiązywaniu własnego problemu informatycznego posłużyć się językiem C++: wykonać analizę problemu, zakodować, uruchomić i przeprowadzić proces debugowania programu.	-	-	+	-	-	-	-	-	-	-	-
M_U003	Umie posłużyć się elementami oferowanymi przez system operacyjny oraz środowisko programowania dla uzyskania przyjaznego dla użytkownika interfejsu programu.	-	-	+	-	-	-	-	-	-	-	-
M_U004	Umie wybrać właściwe dla siebie środowisko programistyczne/uruchomieniowe Java.	-	-	+	-	-	-	-	-	-	-	-
M_U005	Umie tworzyć średnio zaawansowane aplety w Java.	-	-	+	-	-	-	-	-	-	-	-
M_U006	Umie tworzyć proste aplikacje dla urządzeń mobilnych.	-	-	+	-	-	-	-	-	-	-	-
M_U007	Umie wybrać właściwe dla specyficznych w danym miejscu pracy wymagań środowisko programowania obiektowego i potrafi się nim sprawnie posłużyć dla tworzenia i konserwacji oprogramowania użytkowego.	-	-	+	-	-	-	-	-	-	-	-
Kompetencje społeczne: jest gotów do												
M_K001	Potrafi twórczo wykorzystywać aktualnie dostępne techniki tworzenia oprogramowania do własnych potrzeb w pracy zawodowej.	+	-	+	-	-	-	-	-	-	-	-
M_K002	Potrafi poszukiwać i doskonalić się w zakresie nowych sposobów tworzenia przyjaznego użytkownikowi oprogramowania.	-	-	+	-	-	-	-	-	-	-	-

Nakład pracy studenta (bilans punktów ECTS)

Forma aktywności studenta	Obciążenie studenta
Udział w zajęciach dydaktycznych/praktyka	20 godz
Przygotowanie do zajęć	15 godz
przygotowanie projektu, prezentacji, pracy pisemnej, sprawozdania	20 godz
Samodzielne studiowanie tematyki zajęć	30 godz
Sumaryczne obciążenie pracą studenta	85 godz
Punkty ECTS za moduł	3 ECTS

Pozostałe informacje

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

Wykład

Obiektowy paradygmat programowania. Powstanie języka C++, podstawy jego składni i semantyki. Klasy, elementy składowe klas, przeciążenie, polimorfizm, dziedziczenie. Operacje WE/WY biblioteki standardowej oraz interfejsu graficznego. Interferuj przyjazny użytkownikowi.

Filozofia Jawy. B-Code, maszyna wirtualna, rodzaje programów. API Jawy. Środowisko WWW i aplety graficzne. Pełne środowisko graficzne. Programowanie urządzeń telekomunikacyjnych. Rola Jawy w cloud programming.

Kierunki rozwoju języków i systemów programowania obiektowego.

Ćwiczenia laboratoryjne

Środowisko CBuilder oraz Visual C++. Elementy składni języka. Podstawowe biblioteki. Przegląd standardowych kontrolek. Interfejs użytkownika – formy komunikacji. Siatki, wykresy i grafika statyczna oraz animacje.

SDK do Jawy. Java Beans. Biblioteka AWT. Interfejs JDBC. J2ME dla urządzeń mobilnych.

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

Ocena z indywidualnego projektu

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

Wskazana znajomość języka ANSI C

Zalecana literatura i pomoce naukowe

Grebosz J.: Symfonia C++. Editions 2000. Kraków.

Eckel B.: Thinking in C++. Helion. 2002

Heller S.: Whois afraid of Java. 2011

Eckel B.: Thinking in Java. Helion. 2006

Sierra K., Bates B.: Head first Java. Helion. 2004.

Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu

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 universitetu "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 universitetu "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

-