

Nazwa zajęć: Języki i paradygmaty programowania		Programming languages and paradigms	
Kierunek: Informatyka			Obowiązuje od roku ak. 2020/2021
Poziom: I st. inżynierski	Profil: praktyczny	Grupa zajęć: Kierunkowe	
Semestr: II	Forma zaliczenia: Z - zaliczenie na ocenę	Punkty ECTS: 4	Zajęcia do wyboru: Nie
			Język zajęć: polski

Forma zajęć i liczba godzin na studiach stacjonarnych i niestacjonarnych:

Wykład 15 / 8	Ćwiczenia 45 / 24	Suma godzin: 60 / 32
-------------------------	-----------------------------	--------------------------------

Specjalność:

Nazwiska osób odpowiedzialnych za zajęcia:

mgr Marcin Zawadzki

Opis zajęć:

Przedmiot "Języki i paradygmaty programowania" przeznaczony jest dla studentów kierunku Informatyka, którzy przynajmniej w podstawowym zakresie posiadają wiedzę na temat programowania między innymi w takich językach jak: C++, Java, Python itp.. Programowanie jest dziedziną wymagającą abstrakcyjnego, logicznego oraz analitycznego myślenia. Celem przedmiotu jest wykształcenie umiejętności sprawnego posługiwania się dowolnym językiem programowania w sposób, który możliwie najlepiej odzwierciedli rzeczywistość. Całość kursu nastawiona jest na możliwie najprostsze przekazanie wiedzy o tym, czym metody, funkcje, instrukcje warunkowe, pętle itp. oraz jakie w praktyczny sposób wykorzystać zdobytą wiedzę.

Cele dydaktyczne:

Zapoznanie studentów z zagadnieniami dotyczącymi programowania.

Nabywanie przez studenta wiedzy o podstawowych zagadnieniach takich jak instrukcje warunkowe, pętle, metody, funkcje itp.

Nabywanie przez studenta umiejętności tworzenia aplikacji w języku C# z wykorzystaniem konsoli.

Samodzielne radzenie sobie w sytuacji trudnej poprzez analizę problemu oraz pomoc w jego rozwiązaniu przy pomocy dostępnych źródeł informacji.

Metody dydaktyczne:

MP1	wykład informacyjny
MP2	objaśnienie
MP3	praca ze źródłem elektronicznym
MC1	ćwiczenie praktyczne
MS1	metoda sytuacyjna

Metody oceniania:

MO1	kolokwium pisemne
MO2	aktywność w trakcie zajęć

Wykład

W1	Czym jest platforma .NET, czym jest język programowania C# ?
W2	Podstawowe informacje o środowisku oraz zmiennych
W3	Operatory arytmetyczne, przypisania, relacyjne, logiczne oraz bitowe
W4	Instrukcje warunkowe if, else if, else i instrukcje zagnieżdżone.
W5	Pętle for, pętle zagnieżdżone oraz instrukcje break i continue
W6	Instrukcje switch oraz operatory warunkowe
W7	Tablice jednowymiarowe
W8	Pętla foreach, zmienna typu var
W9	Tablice wielowymiarowe oraz wyszczerbione
W10	Funkcje, metody oraz ich przeciążanie

Ćwiczenia

C1	Zapoznanie się ze środowiskiem IDE oraz praca z różnymi typami zmiennych
C2	Zapoznanie się z konsolą oraz pobieraniem danych od użytkownika
C3	Operatory arytmetyczne, przypisania, relacyjne, logiczne oraz bitowe
C4	Instrukcje warunkowe if, else if, else i instrukcje zagnieżdżone.
C5	Pętle for, pętle zagnieżdżone oraz instrukcje break i continue
C6	Instrukcje switch oraz operatory warunkowe
C7	Tablice jednowymiarowe
C8	Pętla foreach, zmienna typu var
C9	Tablice wielowymiarowe oraz wyszczerbione
C10	Funkcje metody oraz ich przeciążanie

Literatura podstawowa

1 C. S. Horstmann: Core Java 2. [T. 1], Podstawy, Gliwice, Helion, 2003
2 C. S. Horstmann: Core Java 2. [T. 2], Techniki zaawansowane, Gliwice, Helion,
3 A. Troelsen: Język C# 2010 i platforma .NET 4, Warszawa, PWN, 2014
4 M. Lis: C# : praktyczny kurs : poznaj tajniki programowania w C#, Gliwice,

Literatura uzupełniająca

Warunki zaliczenia

Warunkiem zaliczenia jest, uzyskanie oceny, co najmniej dostatecznej ze wszystkich kolokwium oraz zaliczenie egzaminu z wiedzy teoretycznej. Ocena końcowa zależy od liczby uzyskanych punktów ze wszystkich składowych oceniania i wynosi:

do 50% - ocena niedostateczna,
51% - 60% - ocena dostateczna,
61% - 70% - ocena dostateczna plus,
71% - 80% - ocena dobra
81% - 90% - ocena dobra plus,
91% - 100% - ocena bardzo dobra.

Przykłady pytań zaliczeniowych

Co to jest instrukcja warunkowa?
 Co to jest metoda?
 Co oznacza przeciążenie funkcji?
 Co to jest zmienna i jakie znasz typy zmiennych?

Obciążenie pracą studenta

Studia stacjonarne/niestacjonarne

Forma pracy studenta	Wykład		Ćwiczenia		Suma	
Zajęcia z bezpośrednim udziałem nauczyciela	15 g	8 g	45 g	24 g	60 g	32 g
Zapoznanie się z literaturą przedmiotu	1 g		1 g	5 g	2 g	5 g
Przygotowanie się do zajęć			20 g	23 g	20 g	23 g
Przygotowanie się do kolokwium						
Realizacja zadanych ćwiczeń i zadań			15 g	30 g	15 g	30 g
Przygotowanie sprawozdania z ćwiczeń						
Przygotowanie projektu / pracy			3 g	10 g	3 g	10 g
Przygotowanie się i udział w egzaminie						
	16 g	8 g	84 g	92 g	100 g	100 g

Efekty uczenia się	KEK	Treści kształcenia	Metody dydaktyczne	M. oceniania
rozumie i opisuje podstawowe cechy systemu opartego o programowanie obiektowe.	K_W03	W7-W8 C3-C6	MP1-MP3	MO1
Korzysta z różnych źródeł w celu porównania różnych rozwiązań i wyboru najkorzystniejszego.	K_U01	W1-W12 C3-C13	MP1-MP3	MO2
potrafi dobrać właściwe polecenia oraz narzędzia do realizacji typowych zadań.	K_U04	W1-W12 C3-C13	MP1-MP3	MO2
jest gotów do rozwiązywania problemów praktycznych, korzystając samodzielnie z dostępnych materiałów lub korzystając z konsultacji wykładowcy oraz przygotowuje się do właściwej realizacji zadań podczas testów	K_K02	W1-W12 C1-C15	MP1-MP3	MO2
jest gotów do dzielenia się swoją wiedzą i umiejętnościami na rzecz środowiska grupy studenckiej, publikując na forum dyskusyjnym wpisy z rozwiązaniami zadań problemowych	K_K03	W1-W12 C1-C15	MP1-MP3	MO2
docenia zalety i dostrzega wady wolnego oprogramowania oraz uzmysławia sobie społeczną rolę programistów w rozpowszechnianiu wykorzystania komputerów przez użytkowników	K_K05	W1-W12 C1-C15	MP1-MP3	MO2
docenia zalety i dostrzega wady wolnego oprogramowania oraz uzmysławia sobie społeczną rolę programistów w rozpowszechnianiu wykorzystania komputerów przez użytkowników	U_K07	W1-W12 C1-C15	MP1-MP3	MO2