

KARTA PRZEDMIOTU

Nazwa przedmiotu	Programowanie w języku C#
------------------	----------------------------------

1. USYTUOWANIE PRZEDMIOTU W SYSTEMIE STUDIÓW

1.1. Kierunek studiów	Informatyka
1.2. Forma i ścieżka studiów	Stacjonarne/Niestacjonarne
1.3. Poziom kształcenia	Studia I stopnia
1.4. Profil studiów	Praktyczny

1.5. Wydział	Wydział Nauk Technicznych
1.6. Specjalność	Nie dotyczy
1.7. Koordynator przedmiotu	dr inż. Bogusław Oleksiejuk

2. OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

2.1. Przynależność do grupy przedmiotu	praktycznych
2.2. Liczba ECTS	3
2.3. Język wykładów	polski
2.4. Semestry, na których realizowany jest przedmiot	5
2.5. Kryterium doboru uczestników zajęć	brak

3. EFEKTY UCZENIA SIĘ I SPOSÓB PROWADZENIA ZAJĘĆ

3.1. Cele przedmiotu

Lp.	Cele przedmiotu
C1	Zapoznanie studentów z podstawowymi cechami, typami i konstrukcjami języka C#
C2	Przekazanie studentom umiejętności wykorzystania zasad programowania obiektowego w programowaniu przy użyciu języka C#
C3	Nabycie przez studentów umiejętności wykorzystania kolekcji, typów generycznych oraz typów delegatów w programowaniu przy użyciu języka C#

3.2. Przedmiotowe efekty uczenia się, z podziałem na WIEDZĘ, UMIEJĘTNOŚCI i KOMPETENCJE, wraz z odniesieniem do kierunkowych efektów uczenia się

Lp.	Opis przedmiotowych efektów uczenia się	Odniesienie do kierunkowych efektów uczenia się (symbole)	Sposób realizacji (X)			
			ST		NST	
			Zajęcia na Uczelni	Obowiązkowe/dodatkowe* zajęcia na platformie	Zajęcia na Uczelni	Obowiązkowe/dodatkowe* zajęcia na platformie
Po zaliczeniu przedmiotu student w zakresie WIEDZY zna i rozumie						
W1	Cechy typy i konstrukcje języka.	INF_W05, INF_W06	x	x	x	x
W2	Podstawowe cech programowania obiektowego w kontekście języka c#.	INF_W05, INF_W06	x	x	x	x
W3	Typy generyczne, delegaty oraz wyrażenia lambda.	INF_W05, INF_W06	x	x	x	x
Po zaliczeniu przedmiotu student w zakresie UMIEJĘTNOŚCI potrafi						
U1	Używać typów, definiować struktury i klasy.	INF_U05, INF_U06, INF_U07, INF_U08	x	x	x	x
U2	Pisać programy, które korzystając z kolekcji.	INF_U05, INF_U06, INF_U07, INF_U08	x	x	x	x
U3	Pisać programy z zastosowaniem abstrakcji, dziedziczenia i polimorfizmu.	INF_U05, INF_U06, INF_U07, INF_U08	x	x	x	x
U4	Obsługiwać błędy, które mogą wystąpić w programach	INF_U05, INF_U06, INF_U07, INF_U08	x	x	x	x
U5	Korzystać z delegatów, zdarzeń i wyrażen lambda.	INF_U05, INF_U06, INF_U07, INF_U08	x	x	X	x
Po zaliczeniu przedmiotu student w zakresie KOMPETENCJI SPOŁECZNYCH jest gotów do						
K1	Rozwijania swoich kompetencji zawodowych	INF_K01	x	x	x	x
K2	Pracy w grupie, definiowania celu działania i weryfikowania osiągnięcia tego celu	INF_K05	x	x	x	x

3.3. Formy zajęć dydaktycznych i ich wymiar godzinowy - Studia stacjonarne (ST), Studia niestacjonarne (NST)

Ścieżka	Wykład	Ćwiczenia	Projekt	Warsztat	Laboratorium	Seminarium	Lektorat	Obowiązkowe/dodatkowe* zajęcia prowadzone z wykorzystaniem metod i technik kształcenia na odległość w formie	Inne	Punkty ECTS
ST	20				40					3
NST	10				20					3

3.4. Treści kształcenia (oddzielnie dla każdej formy zajęć: (W, ĆW, PROJ, WAR, LAB, LEK, INNE). Należy zaznaczyć (X), w jaki sposób dane treści będą realizowane (zajęcia na uczelni lub obowiązkowe / dodatkowe zajęcia na platformie e-learningowej prowadzone z wykorzystaniem metod i technik kształcenia na odległość)

RODZAJ ZAJĘĆ: WYKŁAD

Lp.	Treść zajęć	Sposób realizacji			
		ST		NST	
		ZAJĘCIA NA UCZELNI	OBYWIAZKOWE / DODATKOWE* † ZAJĘCIA NA PLATFORMIE	ZAJĘCIA NA UCZELNI	OBYWIAZKOWE / DODATKOWE* ZAJĘCIA NA PLATFORMIE
1.	Podstawowe typy, konstrukcje i instrukcje	+	+	+	+
2.	Struktury, klasy, interfejsy, kolekcje generyczne i niegeneryczne	+	+	+	+
3.	Obsługa wyjątków i konwersja typów	+	+	+	+
4.	Abstrakcja, dziedziczenie, enkapsulacja i polimorfizm w języku C#	+	+	+	+
5.	Typy generyczne, delegaty, zdarzenia, wyrażenia lambda	+	+	+	+

* Niepotrzebne usunąć

* Niepotrzebne usunąć

RODZAJ ZAJĘĆ: LABORATORIUM

Lp.	Treść zajęć	Sposób realizacji			
		ST		NST	
		ZAJĘCIA NA UCZELNI	OBOWIĄZKOWE / DODATKOWE* ZAJĘCIA NA PLATFORMIE	ZAJĘCIA	OBOWIĄZKOWE / DODATKOWE* ZAJĘCIA NA PLATFORMIE
1.	Podstawowe instrukcje sterujące	+	+	+	+
2.	Kolekcje generyczne	+	+	+	+
3.	Obsługa wyjątków i konwersja typów	+	+	+	+
4.	Struktury, klasy, interfejsy	+	+	+	+
5.	Abstrakcja, dziedziczenie i enkapsulacja	+	+	+	+
6.	Polimorfizm	+	+	+	+
7.	Delegaty, typy Action<T>, Func<T,T>, zdarzenia, wyrażenia lambda	+	+	+	+

+

3.5. Metody weryfikacji efektów uczenia się (wskazanie i opisanie metod prowadzenia zajęć oraz weryfikacji osiągnięcia efektów uczenia się, np. debata, case study, przygotowania i obrony projektu, złożona prezentacja multimedialna, rozwiązywanie zadań problemowych, symulacje sytuacji, wizyta studyjna, gry symulacyjne + opis danej metody):

1. Wykład w postaci prezentacji multimedialnej z przykładami programów oraz elementami programowania „na żywo”.
2. Laboratorium przykłady problemów do rozwiązania z elementami programowania „na żywo”, zlecenie zadań do samodzielnego wykonania podczas zajęć, odnotowywanie aktywności poszczególnych studentów.
3. Wykonywanie zadań zaliczeniowych oraz odpowiedzi pisemne do zadań opisowych

3.6. Kryteria oceny osiągniętych efektów uczenia się

Efekt uczenia się	Na ocenę 3 lub „zal.” student zna i rozumie/potrafi/jest gotów do	Na ocenę 4 student zna i rozumie/potrafi/jest gotów do	Na ocenę 5 student zna i rozumie/potrafi/jest gotów do
W1	Zna podstawowe cechy, typy i konstrukcje języka.	Zna najważniejsze cechy, typy i konstrukcje języka.	Zna większość cech, typów i konstrukcji języka.
W2	Zna co najmniej część podstawowych cech programowania obiektowego w kontekście języka c#.	Zna najważniejsze z podstawowych cech programowania obiektowego w kontekście języka c#.	Zna wszystkie podstawowe cechy programowania obiektowego w kontekście języka c#.
W3	Zna typy generyczne i delegaty.	Zna typy generyczne i delegaty oraz wyrażenia lambda.	Zna typy generyczne i delegaty oraz wyrażenia lambda.

U1	Potrafi używać typów, definiować klasy.	Potrafi używać typów, definiować struktury i klasy.	Potrafi używać typów, definiować struktury i klasy w szerokim zakresie.
U2	Pisać programy, które korzystają z kolekcji.	Pisać programy, które korzystają z kolekcji.	Pisać programy, które korzystają z kolekcji.
U3	Pisać programy z zastosowaniem abstrakcji, dziedziczenia i polimorfizmu.	Pisać programy z zastosowaniem abstrakcji, dziedziczenia i polimorfizmu.	Pisać programy z zastosowaniem abstrakcji, dziedziczenia i polimorfizmu.
U4	Obsługiwać błędy, które mogą wystąpić w programach	Obsługiwać błędy, które mogą wystąpić w programach	Obsługiwać błędy, które mogą wystąpić w programach
U5	Korzystać z delegatów.	Korzystać z delegatów i zdarzeń.	Korzystać z delegatów, zdarzeń i wyrażeń lambda.
K1	Rozumie przynajmniej częściowo potrzebę rozwijania swoich kompetencji zawodowych	Rozumie potrzebę rozwijania swoich kompetencji zawodowych i przynajmniej częściowo potrafi zaplanować swój rozwój osobisty.	Rozumie potrzebę rozwijania swoich kompetencji zawodowych i potrafi zaplanować swój rozwój osobisty.
K2	Rozumie potrzebę pracy w grupie, potrafi przynajmniej częściowo zdefiniować cel działania.	Rozumie potrzebę pracy w grupie, potrafi zdefiniować cel działania i jest w stanie przynajmniej częściowo zweryfikować osiągnięcie tego celu	Rozumie potrzebę pracy w grupie, potrafi zdefiniować cel działania i jest w stanie zweryfikować osiągnięcie tego celu

3.7. Zalecana literatura

Podstawowa

1. C#. Programowanie. Wydanie VI, ISBN 9788324637010, Ian Griffiths, Matthew Adams, Jesse Liberty
2. Visual Studio 2013. Podręcznik programowania w C# z zadaniami, ISBN 9788324668564, Jacek Matulewski

Dodatkowa

1. C# 7.1 i .NET Core 2.0 dla programistów aplikacji wieloplatformowych ISBN 978-83-283-4450-1 Mark J. Price ; tł. Wojciech Moch. Gliwice : Helion, 2019.

4. NAKŁAD PRACY STUDENTA - BILANS PUNKTÓW ECTS

Rodzaje aktywności studenta	Obciążenie studenta	
	studia ST	studia NST
Zajęcia wymagające bezpośredniego kontaktu studenta z nauczycielem akademickim w siedzibie uczelni		
Zajęcia przewidziane planem studiów	60	30
Konsultacje dydaktyczne (mini. 10% godz. przewidzianych na każdą formę zajęć)	6	6

Praca własna studenta		
Przygotowanie bieżące do zajęć, przygotowanie prac projektowych/prezentacji/itp.	8	16
Przygotowanie do zaliczenia zajęć	16	28
SUMARYCZNE OBCIĄŻENIE GODZINOWE STUDENTA	90	80
Liczba punktów ECTS	3	3

Data ostatniej zmiany	04.03.2021
Zmiany wprowadził	dr inż. Bogusław Oleksiejuk
Zmiany zatwierdził	dr inż. Bogusław Oleksiejuk