

Karta opisu zajęć - Sylabus

Państwowa Wyższa Szkoła Techniczno-Ekonomiczna im. ks. Bronisława Markiewicza w Jarosławiu

I. INFORMACJE PODSTAWOWE

Nazwa zajęć: Języki programowania wysokiego poziomu	Cykl kształcenia: 2021/2022	Data aktualizacji sylabusa: 01.10.2021
Nazwa kierunku studiów, poziom i profil kształcenia: Automatyka i Elektronika Praktyczna, studia pierwszego stopnia, profil praktyczny		
Język wykładowy: polski	Rodzaj zajęć: Przedmiot kierunkowy	
Rok studiów: 1	Semestr: 2	
Liczba punktów ECTS przypisana zajęciom: 5	Koordynator zajęć Imię, nazwisko, tytuł/stopień naukowy, adres e-mail: Janusz Bytnar, dr inż., janusz.bytnar@pwste.edu.pl	
Jednostka organizacyjna: Instytut Inżynierii Technicznej	Prowadzący zajęcia Imię, nazwisko, tytuł/stopień naukowy, adres e-mail: Janusz Bytnar, dr inż., janusz.bytnar@pwste.edu.pl	

FORMA PROWADZENIA ZAJĘĆ I LICZBA GODZIN

Ogólna liczba godzin zajęć dydaktycznych na studiach stacjonarnych i niestacjonarnych z podziałem na formy:

Studia stacjonarne		Studia niestacjonarne	
Wykład:	30	Wykład:	
Ćwiczenia:		Ćwiczenia:	
Laboratorium:		Laboratorium:	
Lektorat:		Lektorat:	
Projekt:		Projekt:	
Zajęcia praktyczne:	45	Zajęcia praktyczne:	
Seminarium:		Seminarium:	
Zajęcia terenowe:		Zajęcia terenowe:	
Praktyki:		Praktyki:	
Inna forma (jaka):		Inna forma (jaka):	
RAZEM:	75	RAZEM:	

II. INFORMACJE SZCZEGÓŁOWE

Wymagania wstępne i dodatkowe:

wymagania formalne - student powinien być zarejestrowany na semestr drugi
wymagania wstępne – Podstawy programowania.

Cel (cele) kształcenia dla zajęć:

Celem przedmiotu jest omówienie współczesnych technik programowania, w tym programowania strukturalnego, obiektowego, obiektowo orientowanego i funkcjonalnego na bazie wybranych języków programowania ze szczególnym uwzględnieniem języków: C++, C# i Java.

Efekty uczenia się określone dla zajęć

<p>Efekty uczenia się określone dla zajęć w kategorii wiedza, umiejętności oraz kompetencje społeczne oraz metody weryfikacji efektów uczenia się</p> <p>UWAGA: Dzielimy efekty uczenia się określone dla zajęć na kategorie wiedzy, umiejętności i kompetencji społecznych. Określone dla zajęć efekty uczenia się nie muszą obejmować wszystkich trzech kategorii i zależą od formy zajęć.</p>				
Symbol efektów uczenia się określonego dla zajęć*		Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się, student w kategorii:		
Wiedzy - zna i rozumie				
M_01	Zna składnię i semantykę języków programowania, sposoby przekazywania parametrów do podprogramów, potrafi zdefiniować abstrakcyjne typy danych oraz wyjaśnić zasady przeciążania operatorów.			
M_02	Zna i rozumie zastosowanie metodologii programowania obiektowego podczas rozwiązywania problemów informatycznych.			
M_03	Ma uporządkowaną wiedzę z zakresu języków programowania wysokiego poziomu, zna zasady doboru języka programowania do rozwiązywania problemów w zakresie oprogramowania sprzętu i usług; rozumie metody specyfikowania podstawowych wymagań w zakresie oprogramowania.			
Umiejętności - potrafi				
M_04	Potrafi pracować indywidualnie i w zespole; umie oszacować czas potrzebny na realizację zleconego zadania; potrafi opracować i zrealizować harmonogram prac zapewniający dotrzymanie terminów.			
M_05	Ma umiejętność tworzenia projektów programistycznych w oparciu o języki programowania wysokiego poziomu.			
M_06	Potrafi korzystać z dokumentacji i specyfikacji technicznych w celu dobrania odpowiednich parametrów i komponentów projektowanego systemu.			
M_07	Potrafi sformułować algorytm, posługuje się językami programowania wysokiego poziomu oraz odpowiednimi narzędziami informatycznymi do opracowania programów komputerowych.			
Kompetencje społecznych - jest gotów do				
M_08	Rozumie potrzebę uczenia się przez całe życie.			
M_09	Służy wsparciem i pomocą członkom zespołu w trudnych i stresujących sytuacjach.			
<p>UWAGA! Zaleca się, aby w zależności od liczby godzin zajęć, liczba efektów uczenia się zawierała się w przedziale: 3-7, ale są to wartości umowne.</p>				
TREŚCI PROGRAMOWE I ICH ODNIESIENIE DO EFEKTÓW UCZENIA SIĘ OKREŚLONYCH DLA ZAJĘĆ				
Treści programowe (uszczegółowione, zaprezentowane z podziałem na poszczególne formy zajęć, tj. wykład, ćwiczenia, laboratoria, projekty, seminaria i inne):				
Symbol treści programowych	Opis treści programowych	Forma zajęć	Liczba godzin	Odniesienie do efektów uczenia się określonych dla zajęć (symbol efektów uczenia się)
		wykład		

TP-01	Omówienie tematyki przedmiotu, literatury, form i zasad zaliczenia. Pojęcia ogólne: opis składni i semantyki języków programowania – C++, C# i Java. Omówienie aktualnych trendów rozwojowych w poszczególnych językach programowania.		2	M_01
TP-02	Wykorzystanie abstrakcyjnych typów danych. Możliwości wykorzystania przeciążania operatorów dla własnych typów danych. Przekazywanie argumentów do funkcji w poszczególnych językach programowania.		2	M_01, M_02
TP-03	Mechanizmy zarządzania pamięcią (odśmiecania pamięci podczas działania programu) w języku C++, C# oraz Java. Przykłady programów. Omówienie cech charakterystycznych programowania imperatywnego.		2	M_01, M_02
TP-04	Przedstawienie cech charakterystycznych programowania obiektowego dla poszczególnych języków C++, C# i Java. Definiowanie klas: prawa dostępu, konstruktor, destruktor, lista inicjacyjna, pola i funkcje statyczne, funkcje zaprzyjaźnione, tworzenie obiektów, dostęp do obiektów. Omówienie przykładowych programów z języków: C++, Java, C#.		5	M_01, M_02, M_03
TP-05	Omówienie możliwości dziedziczenia i polimorfizmu w językach obiektowych. Pojęcie klasy bazowej i pochodnej, przesłanianie składowych, wiązanie statyczne i dynamiczne, funkcje wirtualne, klasy abstrakcyjne, interfejsy. Wykład z komentarzem i przykładami w języku C++, Java, C#.		5	M_01, M_02, M_03
TP-06	Przykłady projektów realizujących zadania programistyczne wykorzystujących możliwości programowania obiektowego. Omówienie przykładów w wybranych językach C# i Java.		4	M_01, M_02, M_03, M_08

TP-07	<p>Programowanie graficznych interfejsów użytkownika. Przegląd komponentów oraz ich właściwości. Okna i rozmieszczenie komponentów (managery rozkładu). Zalety i wady ręcznego programowania interfejsu użytkownika. Możliwości szybkiego tworzenia graficznych interfejsów użytkownika w środowisku Eclipse, Microsoft Visual Studio.</p>		5	M_01, M_02, M_03, M_08
TP-08	<p>Omówienie tematu obsługi zdarzeń w aplikacjach okienkowych – interakcja użytkownika z aplikacją. Rodzaje zdarzeń, obsługa zdarzeń. Omówienie przykładowych aplikacji w językach C# i Java.</p>		5	M_01, M_02, M_03, M_08
		zajęcia praktyczne		
TP-09	<p>Zajęcia praktyczne – zapoznanie ze środowiskiem programistycznym Microsoft Visual Studio oraz Eclipse. Tworzenie nowych projektów. Pisanie kodów źródłowych w celu powtórzenia podstawowych typów danych, funkcji, pętli, tablic.</p>		4	M_01, M_02, M_03, M_04, M_05, M_06, M_07, M_08, M_09
TP-10	<p>Tworzenie i usuwanie tablic dynamicznych w języku C++. Pisanie programów oraz testowanie mechanizmu automatycznego zarządzania pamięcią w językach C# i Java. Tworzenie przykładowych programów wykorzystujących obiekty. Tworzenie schematów klas, konstruktorów, destruktorów. Pisanie funkcji składowych. Przekazywanie obiektów do funkcji w postaci argumentu. Funkcje zaprzyjaźnione oraz ich możliwości. Testowanie możliwości dostępu do składników obiektu, uruchamianie funkcji prywatnych. Przykłady w języku C i Java.</p>		10	M_01, M_02, M_03, M_04, M_05, M_06, M_07, M_08, M_09

TP-11	Zajęcia praktyczne - tworzenie klas dziedziczących z wcześniej utworzonych, tworzenie klas abstrakcyjnych i interfejsów oraz ich wykorzystanie w nowych klasach – pisanie przykładowych programów w języku C i Java. Dostęp do składników obiektów złożonych. Pisanie oraz testowanie funkcji wirtualnych.		6	M_01, M_02, M_03, M_04, M_05, M_06, M_07, M_08, M_09
TP-12	Tworzenie aplikacji Windows Forms – tworzenie prostych interfejsów użytkownika w C++ i C# – testowanie aplikacji. Techniki zarządzania układem graficznym. Układ komponentów. Obsługa zdarzeń. Komponenty – wprowadzania tekstu, dokonywania wyboru. Okna dialogowe. Budowa menu.		8	M_01, M_02, M_03, M_04, M_05, M_06, M_07, M_08, M_09
TP-13	Tworzenie prostych aplikacji i apletów w języku Java. Komponenty AWT i Swing. Układ graficzny aplikacji. Obsługa zdarzeń – interfejs nasłuchu zdarzeń. Pola wyboru, okna dialogowe.		8	M_01, M_02, M_03, M_04, M_05, M_06, M_07, M_08, M_09
TP-14	Realizacja aplikacji okienkowej w języku C# lub Java operującej na danych w postaci obiektów. Aplikacja wykorzystuje możliwości zapisu i odczytu informacji o obiektach w plikach tekstowych – prosta baza danych.		9	M_01, M_02, M_03, M_04, M_05, M_06, M_07, M_08, M_09

ZALECANA LITERATURA (w tym pozycje w języku obcym)

Literatura podstawowa (powinna być dostępna dla studenta w uczelnianej bibliotece):

1. Grębosz J.: Symfonia C++ standard, Edycja 2015
2. Snell M., Power L.: „Microsoft Visual Studio 2010, Helion, Gliwice, 2011,
3. Schildt H.: „Java : sztuka programowania” , Helion, Gliwice, 2004

Literatura uzupełniająca:

1. Boduch A.: Wstęp do programowania w C#, Helion, Gliwice, 2006
2. Troelsen S.: Język C# i Platforma .NET, Wydawnictwo Naukowe PWN, Warszawa2006

III. INFORMACJE DODATKOWE

Odniesienie efektów uczenia się określonych dla zajęć i treści programowych do form zajęć i metod oceniania

Symbol efektu uczenia się określonego dla zajęć	Symbol treści programowych realizowanych w trakcie zajęć	Formy zajęć i metody dydaktyczne prowadzenia zajęć umożliwiające osiągnięcie założonych efektów uczenia się *	Metody weryfikacji osiągnięcia efektów uczenia się przypisanych do zajęć #
---	--	---	--

	Wiedza	wykład	
M_01	TP_01 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
M_02	TP_02 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
M_03	TP_04 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
	Umiejętności	ćwiczenia, laboratorium, projekt, zajęcia praktyczne	
M_04	TP_09 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
M_05	TP_09 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
M_06	TP_09 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
M_07	TP_09 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
	Kompetencje społeczne	ćwiczenia, laboratorium, projekt, zajęcia praktyczne	
M_08	TP_06 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
M_09	TP_09 – TP_14	Wykład i zajęcia praktyczne – prezentacja, wykład z dyskusją, praktyczna realizacja aplikacji	Test, kolokwium, projekt zaliczeniowy
<p>Metody weryfikacji osiągnięcia efektów uczenia się określonych dla zajęć, powinny być zróżnicowane w zależności od kategorii, tj. inne dla kategorii wiedza i inne dla kategorii umiejętności i kompetencje społeczne.</p> <p>Dla wykładu: * np. wykład podający, wykład problemowy, ćwiczenia oparte na wykorzystaniu różnych źródeł wiedzy # np. egzamin ustny, test, prezentacja, projekt</p> <p>Zaleca się podanie przykładowych zadań (pytań) służących weryfikacji osiągnięcia efektów uczenia się określonych dla zajęć.</p>			

BILANS PUNKTÓW ECTS			
OBCIĄŻENIE PRACĄ STUDENTA (godziny)			
Forma aktywności		Liczba godzin *	
Godziny zajęć (według harmonogramu) z nauczycielem akademickim lub inną osobą prowadzącą zajęcia		75	
Praca własna studenta		75	
SUMA GODZIN:		150	
OBCIĄŻENIE PRACĄ STUDENTA (punkty ECTS)			
		Liczba punktów ECTS	
SUMARYCZNA LICZBA PUNKTÓW ECTS PRZYPISANYCH DO ZAJĘĆ	Praca studenta wymagająca bezpośredniego kontaktu z nauczycielem akademickim lub inną osobą prowadzącą zajęcia	Ogółem: 5	3
	Praca własna studenta		2
* godziny lekcyjne, czyli 1 godz. oznacza 45 min;			
OPIS PRACY WŁASNJE STUDENTA:			
Praca własna studenta musi być precyzyjnie opisana, uwzględniając charakter praktyczny zajęć. Należy podać symbol efektu uczenia się, którego praca własna dotyczy oraz metody weryfikacji efektów uczenia się stosowane w ramach pracy własnej. Przykładowe formy aktywności: (1) przygotowanie do zajęć, (2) opracowanie wyników, (3) czytanie wskazanej literatury, (4) napisanie raportu z zajęć, (5) przygotowanie do egzaminu, opracowanie projektu.			
Samodzielna praca związana z czytaniem wskazanej literatury oraz przygotowanie do zaliczenia końcowego (M_01 – M_03). Samodzielna realizacja ćwiczeń programistycznych na podstawie instrukcji, przygotowanie mini-projektów indywidualnych oraz grupowych (M_04 – M_09).			
KRYTERIA OCENIANIA			
Zajęcia kończą się zaliczeniem na ocenę. Na ocenę dostateczną student zrealizował projekty programistyczne dotyczące opisywanych treści programowych przedmiotu w zakresie podstawowym i potrafi omówić zastosowane techniki. Ponadto w dostatecznym stopniu potrafi wykorzystać przekazaną wiedzę podczas wykładu w realizowanych projektach programistycznych oraz podczas zaliczenia pisemnego. Na ocenę dobrą student zrealizował kompletne projekty programistyczne dotyczące opisywanych treści programowych przedmiotu i potrafi omówić zastosowane techniki oraz uzasadnić ich wybór. Ponadto w dobrym stopniu potrafi wykorzystać przekazaną wiedzę podczas wykładu w realizowanych projektach programistycznych oraz podczas zaliczenia pisemnego. Na ocenę bardzo dobrą student zrealizował kompletne projekty programistyczne dotyczące opisywanych treści programowych przedmiotu z zastosowaniem ponadstandardowych rozwiązań i potrafi omówić zastosowane techniki oraz uzasadnić ich wybór. Ponadto w bardzo dobrym stopniu potrafi wykorzystać przekazaną wiedzę podczas wykładu w realizowanych projektach programistycznych oraz podczas zaliczenia pisemnego.			
INFORMACJA O PRZEWIDYWANEJ MOŻLIWOŚCI WYKORZYSTANIA B-LEARNINGU			
INFORMACJA O PRZEWIDYWANEJ MOŻLIWOŚCI WYKORZYSTANIA E-LEARNINGU			

.....
(data, podpis Koordynatora
odpowiedzialnego za zajęcia)

.....
(data, podpis Dyrektora Instytutu/
Kierownika Jednostki Międzyinstytutowej)

.....
(data, podpis Kierownika Zakładu)

Uwaga:
Karta opisu zajęć (sylabus) musi być dostępna dla studenta.