



### KARTA PRZEDMIOTU

Kod przedmiotu	studia stacjonarne:	<b>Z-ZIP1-U-210</b>
	studia niestacjonarne:	<b>Z-ZIPN1-U-210</b>
Nazwa przedmiotu	<b>Podstawy informatyki</b>	
Nazwa przedmiotu w języku angielskim	<b>The Fundamentals of Computers Science</b>	
Obowiązuje od roku akademickiego	<b>2019/2020</b>	

### USYTUOWANIE PRZEDMIOTU W SYSTEMIE STUDIÓW

Kierunek studiów	<b>ZARZĄDZANIE I INŻYNIERIA PRODUKCJI</b>
Poziom kształcenia	<b>I stopień</b>
Profil studiów	<b>Ogólnoakademicki</b>
Forma i tryb prowadzenia studiów	<b>Studia stacjonarne i niestacjonarne</b>
Zakres	<b>Wszystkie zakresy</b>
Jednostka prowadząca przedmiot	<b>Katedra Technologii Informatycznych</b>
Koordinator przedmiotu	<b>dr inż. Sławomir Koczubiej</b>
Zatwierdził	<b>dr hab. inż. Artur Bartosik, prof. PŚk</b>

### OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	<b>Przedmiot kierunkowy</b>	
Status przedmiotu	<b>Obowiązkowy</b>	
Język prowadzenia zajęć	<b>Polski</b>	
Usytuowanie w planie studiów - semestr	studia stacjonarne	<b>Semestr I</b>
	studia niestacjonarne	<b>Semestr I</b>
Wymagania wstępne	<b>Technologie informacyjne</b>	
Egzamin (TAK/NIE)	<b>NIE</b>	
Liczba punktów ECTS	<b>3</b>	

Forma prowadzenia zajęć		wykład	ćwiczenia	laboratorium	projekt	inne
Liczba godzin w semestrze	studia stacjonarne:	<b>15</b>		<b>30</b>		
	studia niestacjonarne:	<b>9</b>		<b>18</b>		

## EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty uczenia się	Odniesienie do efektów kierunkowych
Wiedza	W01	Zna podstawowe algorytmy obliczeniowe oraz przetwarzania danych.	ZIP1_W05
	W02	Ma wiedzę o prostych i złożonych typach danych (tablica, lista, plik, obiekt).	ZIP1_W05
	W03	Posiada wiedzę o składni, gramatyce oraz instrukcjach wybranego języka programowania, jego podstawowych funkcjach bibliotecznych i wbudowanych.	ZIP1_W05
	W04	Ma podstawową wiedzę o współczesnych aplikacjach sieciowych i internetowych.	ZIP1_W04 ZIP1_W05
	W05	Ma podstawową wiedzę na temat oprogramowania użytkowego do obliczeń naukowych i inżynierskich (CAS)	ZIP1_W04 ZIP1_W05
Umiejętności	U01	Student potrafi implementować proste algorytmy w języku programowania.	ZIP1_U07 ZIP1_U14
	U02	Potrafi rozwiązywać podstawowe problemy z zakresu analizy matematycznej, matematyki finansowej, algebry, podstaw statystyki, prezentować wyniki obliczeń w postaci graficznej wykorzystując pakiet do obliczeń matematyczno-statystycznych.	ZIP1_U07 ZIP1_U14
	U03	Potrafi korzystać z dokumentacji technicznej, podręczników i źródeł internetowych w celu poszerzenia swojej wiedzy o językach programowania i pakietach obliczeniowych.	ZIP1_U07 ZIP1_U14
Kompetencje społeczne	K01	Jest gotów do pracy w zespole podczas rozwiązywania wspólnych zadań. Współdziała z innymi członkami zespołu na różnych etapach rozwiązywania problemu	ZIP1_K04

## TREŚCI PROGRAMOWE

Forma zajęć	Treści programowe
wykład	<ol style="list-style-type: none"><li>1. Wprowadzenie do programowania. Semantyka i syntaktyka języka programowania. Wyrażenia algebraicznych i logiczne. Instrukcje wejścia/wyjścia. Proces translacji oraz uruchamiania programu.</li><li>2. Reprezentacja danych w pamięci komputera. Podstawowe typy danych: liczbowe, znakowe, wyliczeniowe, inne. Proste instrukcje sterujące: warunkowa i wyboru. Iteracyjne instrukcje sterujące – pętle.</li><li>3. Korzystanie z wbudowanych funkcji i bibliotek języka. Zapis programów z użyciem własnych procedur i funkcji. Przekazywanie parametrów do podprogramów. Zasięg zmiennych. Typ plikowy Obsługa różnego rodzaju plików (tekstowe, binarne).</li><li>4. Komputerowe wspomaganie obliczeń inżynierskich – oprogramowanie typu CAS (Computer Algebra System). Wprowadzenie do wybranego oprogramowania</li><li>5. Obliczenia numeryczne i symboliczne z zakresu analizy matematycznej, algebry i statystyki w wybranym systemie CAS. Komunikacja systemu z tekstowymi plikami dyskowymi.</li></ol>

laboratorium	<ol style="list-style-type: none"> <li>1. Definiowanie prostych algorytmów, zapis w różnych notacjach (np. w postaci sieci działań). Proste typy danych (liczbowe, znakowe). Zmienne, operatory i wyrażenia.</li> <li>2. Instrukcja przypisania. Komunikacja z użytkownikiem: instrukcje wejścia/wyjścia. Instrukcje sterujące: instrukcja warunkowa i wyboru. Konwersja typów danych. Korzystanie z funkcji bibliotecznych oraz wbudowanych.</li> <li>3. Iteracyjne instrukcje sterujące – pętle. Programowanie z wykorzystaniem typu tablicowego i typów pochodnych. Definiowanie własnych funkcji i procedur. Zasięg zmiennych. Parametry procedur i funkcji i sposoby ich przekazywania.</li> <li>4. Programowanie z wykorzystaniem plików tekstowych i binarnych.</li> <li>5. Oprogramowanie użytkowe typu CAS – środowisko, zapis wyrażień arytmetycznych i podstawowych funkcji. Generowanie wykresów funkcji</li> <li>6. Oprogramowanie CAS. Operacje na wektorach i macierzach. Rozwiązywanie równań, układów równań, nierówności. Analiza statystyczna.</li> </ol>
--------------	--

### **METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ**

Symbol efektu	Metody sprawdzania efektów kształcenia <i>(zaznaczyć X)</i>					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01			X			
W02			X			
W03			X			
W04			X			
W05			X			
U01			X			
U02			X			
U03			X			
K01						X

### **FORMA I WARUNKI ZALICZENIA**

Forma zajęć	Forma zaliczenia	Warunki zaliczenia
wykład	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów z testu na wykładzie.
laboratorium	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów z kolokwίων w trakcie zajęć.

## NAKŁAD PRACY STUDENTA

Bilans punktów ECTS													
Lp.	Rodzaj aktywności	Obciążenie studenta										Jednostka	
		studia stacjonarne					studia niestacjonarne						
		W	C	L	P	S	W	C	L	P	S		
1.	Udział w zajęciach zgodnie z planem studiów	15		30			9		18				h
2.	Inne (konsultacje, egzamin)	2		2			2		2				h
3.	<b>Razem przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>49</b>					<b>31</b>					h	
4.	<b>Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>2,0</b>					<b>1,2</b>					ECTS	
5.	<b>Liczba godzin samodzielnej pracy studenta</b>	<b>26</b>					<b>44</b>					h	
6.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy</b>	<b>1,0</b>					<b>1,8</b>					ECTS	
7.	<b>Nakład pracy związany z zajęciami o charakterze praktycznym</b>	<b>50</b>					<b>50</b>					h	
8.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym</b>	<b>2,0</b>					<b>2,0</b>					ECTS	
9.	<b>Sumaryczne obciążenie pracą studenta</b>	<b>75</b>					<b>75</b>					h	
10.	<b>Punkty ECTS za moduł</b> <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	<b>3</b>										ECTS	

## LITERATURA

1. Lembas J., Kawa R., Wstęp do informatyki, PWN, Warszawa 2017.
2. Wróblewski P., Algorytmy, struktury danych i techniki programowania, Helion, Gliwice 2015.
3. Krup F., Algorytmy. Struktury danych i złożoność obliczeniowa, Helion, Gliwice 2022.
4. Lutz M., Python. Wprowadzenie, Helion, Gliwice 2020.
5. Dawson M., Python dla każdego. Podstawy programowania, Helion, Gliwice 2021,