



## IV. Opis programu studiów

### 3. KARTA PRZEDMIOTU

Kod przedmiotu	Z-ZIPN1-U-508a
Nazwa przedmiotu	Języki programowania – C++
Nazwa przedmiotu w języku angielskim	Programming languages – C++
Obowiązuje od roku akademickiego	2019/2020

#### USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	ZARZĄDZANIE I INŻYNIERIA PRODUKCJI
Poziom kształcenia	I stopień
Profil studiów	Ogólnoakademicki
Forma i tryb prowadzenia studiów	Studia niestacjonarne
Zakres	Wszystkie zakresy
Jednostka prowadząca przedmiot	Katedra Informatyki i Matematyki Stosowanej
Koordinator przedmiotu	dr hab. Marzena Nowakowska
Zatwierdził	dr hab. inż. Artur Bartosik, prof. PŚk

#### OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	Przedmiot kierunkowy
Status przedmiotu	Obowiązkowy
Język prowadzenia zajęć	Polski
Usytuowanie modułu w planie studiów - semestr	Semestr V
Wymagania wstępne	Podstawy informatyki, Informatyka - programowanie
Egzamin (TAK/NIE)	NIE
Liczba punktów ECTS	3

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	9		18		

## EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Student zna i rozumie zasady funkcjonowania programów komputerowych i zarządzania nimi w środowisku Windows.	ZIP_W05
	W02	Student ma podstawową wiedzę na temat typów danych i doboru tych typów do rozwiązania określonego zadania programistycznego.	ZIP_W05
	W03	Student rozumie modułową budowę programu komputerowego oraz potrzebę wykorzystania w nim wbudowanych funkcji C++ w środowisku IDE i własnych funkcji programisty.	ZIP_W05
Umiejętności	U01	Student potrafi zaprojektować i zbudować aplikację okienkową, korzystając z gotowych obiektów systemu programistycznego C++ w IDE oraz własnych rozwiązań programistycznych.	ZIP_U07
	U02	Student potrafi dokonać analizy prostego kodu źródłowego oraz wprowadzić modyfikacje w treści istniejącego programu.	ZIP_U07
	U03	Student posiada umiejętności definiowania własnych funkcji programisty oraz wykorzystania ich w tworzonej przez siebie aplikacji komputerowej.	ZIP_U07
	U04	Student posiada umiejętności konstruowania algorytmów służących rozwiązaniu różnych zadań programistycznych zgodnie z zasadami logiki uniwersalnej.	ZIP_U07
Kompetencje społeczne	K01	Student rozumie potrzebę stałego stosowania i uzupełniania wiedzy z zakresu działania algorytmicznego.	ZIP-K01
	K02	Student potrafi pracować samodzielnie i w grupie (przyjmując w niej różne role).	ZIP-K04

## TREŚCI PROGRAMOWE

Forma zajęć	Treści programowe
wykład	1. Środowisko pracy IDE. Struktura programu w języku C++. Struktura aplikacji C++ w IDE. Tworzenie kodu wynikowego. Paleta komponentów, komunikacja z użytkownikiem.
	2. Podstawowe instrukcje języka. Typy proste. Konwersja typów. Wybrane operatory, ich priorytet i łączność.
	3. Algorytmy przetwarzania iteracyjnego. Tablice i instrukcje pętli.
	4. Typy wskaźnikowe. Operatory wskazania i wyluskania. Arytmetyka adresowa. Praca z tablicami. Sterowanie w programie z wykorzystaniem instrukcji pętli.
	5. Definiowanie funkcji i przekazywanie parametrów. Elementy programowania obiektowego na przykładzie komponentów dostępnych w IDE. Właściwości i metody klas jako narzędzia działania na obiektach.
	6. Klasa łańcuchów tekstowych; właściwości i metody. Przetwarzanie tekstów.
	7. Organizacja dostępu do pliku tekstowego z wykorzystaniem komponentów dialogowych. Współpraca aplikacji z plikiem tekstowym. Transmisja danych między aplikacją i plikiem tekstowym.
	8. Organizacja dostępu do pliku tekstowego z wykorzystaniem komponentów dialogowych. Współpraca aplikacji z plikiem tekstowym. Transmisja danych między aplikacją i plikiem tekstowym.

laboratorium	1. Struktura projektu w środowisku IDE dla C++. Komunikacja aplikacji z użytkownikiem. Kolekcja obiektów standardowych.
	2. Konwersja typów i operacje arytmetyczne. Kolejność działań w programie. Sterowanie w programie. Wprowadzanie danych i wyprowadzanie wyników.
	3. Algorytmy obliczeniowe. Przetwarzanie iteracyjne danych numerycznych. Implementacja algorytmów obliczeniowych w aplikacji C++ utworzonej w IDE. Tablice i instrukcje pętli: wyznaczanie statystyk z tablic numerycznych.
	4. Przetwarzanie iteracyjne tablic z wykorzystaniem klasy typów tekstowych.
	5. Funkcja użytkownika jako składowa klasy i jako funkcja zewnętrzna. Zmienne globalne. Przekazywanie parametrów funkcji.
	6. Przetwarzanie ciągów znaków. Iteracyjne przetwarzanie tekstów: statystyki, wyszukiwanie i modyfikowanie tekstów.
	7. Organizacja dostępu do plików tekstowych. Transmisja danych między pamięcią dyskową i pamięcią operacyjną (struktury tablicowe i wizualne komponenty formularza). Tworzenie aplikacji współpracującej z plikiem tekstowym.

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

Symbol efektu	Metody sprawdzania efektów kształcenia					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01			X			X
W02			X			X
W03			X			X
U01			X			X
U02			X			X
U03			X			X
U04			X			X
K01						X
K02						X

### **FORMA I WARUNKI ZALICZENIA**

Forma zajęć	Forma zaliczenia	Warunki zaliczenia
wykład	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów ze sprawdzianu końcowego.
ćwiczenia	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów z kolokwium zaliczeniowego.

## NAKŁAD PRACY STUDENTA

Bilans punktów ECTS							
Lp.	Rodzaj aktywności	Obciążenie studenta					Jednostka
		W	C	L	P	S	
1.	Udział w zajęciach zgodnie z planem studiów	9		18			h
2.	Inne (konsultacje, egzamin)	2		2			h
3.	<b>Razem przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>31</b>					h
4.	<b>Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>1,2</b>					ECTS
5.	<b>Liczba godzin samodzielnej pracy studenta</b>	<b>44</b>					h
6.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy</b>	<b>1,8</b>					ECTS
7.	<b>Nakład pracy związany z zajęciami o charakterze praktycznym</b>	<b>50</b>					h
8.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym</b>	<b>2</b>					ECTS
9.	<b>Sumaryczne obciążenie pracą studenta</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. Daniluk A. (2003), *C++ Builder. Ćwiczenia praktyczne*, Helion, Gliwice.
2. Daniluk A. (2006), *C++Builder Borland Developer Studio 2006. Kompendium programisty*, Helion, Gliwice..
3. Grębosz J. (2018), *Opus magnum C++11. Programowanie w języku C++*, Helion, Gliwice.
4. Lis M. (2012), *C#. Praktyczny kurs*, Helion, Gliwice.
5. Matulewski J. (2018), *Visual Studio 2017. Tworzenie aplikacji Windows w języku C#*, Helion, Gliwice.
6. Stroustrup B. (2014), *Język C++. Kompendium wiedzy*, Helion, Gliwice.
7. Wróblewski P. (2019), *Algorytmy, struktury danych i techniki programowania*, Helion, Gliwice.