



KARTA PRZEDMIOTU

Kod przedmiotu	studia stacjonarne:	Z-ZIP1-U-508a
	studia niestacjonarne:	Z-ZIPN1-U-508a
Nazwa przedmiotu	Języki programowania – C++	
Nazwa przedmiotu w języku angielskim	Programming Languages – C++	
Obowiązuje od roku akademickiego	2019/2020	

USYTUOWANIE PRZEDMIOTU W SYSTEMIE STUDIÓW

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

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	Przedmiot kierunkowy	
Status przedmiotu	Wybieralny	
Język prowadzenia zajęć	Polski	
Usytuowanie w planie studiów - semestr	studia stacjonarne	Semestr V
	studia niestacjonarne	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	studia stacjonarne:	15		30		
	studia niestacjonarne:	9		18		

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty uczenia się	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	<ol style="list-style-type: none">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 wyłuskania. Arytmetyka5. adresowa. Praca z tablicami. Sterowanie w programie z wykorzystaniem instrukcji pętli.6. Definiowanie funkcji i przekazywanie parametrów. Elementy programowania obiektowego na przykładzie komponentów dostępnych w IDE. Właściwości i metody klas jako narzędzia działania na obiektach.7. Klasa łańcuchów tekstowych; właściwości i metody. Przetwarzanie tekstów.8. Organizacja dostępu do pliku tekstowego z wykorzystaniem komponentów dialogowych. Współpraca aplikacji z plikiem tekstowym. Transmisja danych między aplikacją i plikiem tekstowym.

laboratorium	<ol style="list-style-type: none"> 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 <i>(zaznaczyć X)</i>					
	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.
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.	Razem przy bezpośrednim udziale nauczyciela akademickiego	49					31					h
4.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	2,0					1,2					ECTS
5.	Liczba godzin samodzielnej pracy studenta	26					44					h
6.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	1,0					1,8					ECTS
7.	Nakład pracy związany z zajęciami o charakterze praktycznym	50					50					h
8.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	2,0					2,0					ECTS
9.	Sumaryczne obciążenie pracą studenta	75					75					h
10.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	3										ECTS

LITERATURA

1. Allain A. (2014), C++ przewodnik dla początkujących, Helion, Gliwice.
2. Čukić I. (2019), Programowanie funkcyjne w C++: tworzenie lepszych aplikacji, Helion, Gliwice.
3. Grębosz J. (2018), Opus magnum C++11. Programowanie w języku C++, Helion, Gliwice.
4. Stroustrup B. (2014), Język C++. Kompendium wiedzy, Helion, Gliwice.
5. Swart B., Cashman M., Gustavson P., Hollingworth J., *Borland C++Builder 6 Developer's Guide*, SAMS, dostępne ze strony: http://140.129.118.16/~richwang/ktlan/BCPPB6_Book_unlocked.pdf.
6. Wróblewski P. (2019), Algorytmy, struktury danych i techniki programowania, Helion, Gliwice.