



Karta przedmiotu

Nazwa i kod przedmiotu	Programming, PG_00046329						
Kierunek studiów	Elektronika i telekomunikacja (studia w jęz. angielskim), Informatyka (studia w jęz. angielskim), Automatyka, cybernetyka i robotyka (studia w jęz. angielskim)						
Data rozpoczęcia studiów	październik 2021 r.	Rok akademicki realizacji przedmiotu			2021/2022		
Poziom kształcenia	II stopnia	Grupa zajęć			Grupa zajęć obowiązkowych z zakresu kierunku studiów		
Forma studiów	stacjonarne	Sposób realizacji			na uczelni		
Rok studiów	1	Język wykładowy			angielski		
Semestr studiów	1	Liczba punktów ECTS			5.0		
Profil kształcenia	ogólnoakademicki	Forma zaliczenia			egzamin		
Jednostka prowadząca	Wydział Elektroniki -> Telekomunikacji i Informatyki -> Katedra Algorytmów i Modelowania Systemów						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot	dr Marcin Jurkiewicz					
	Prowadzący zajęcia z przedmiotu	dr Marcin Jurkiewicz					
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	15.0	10.0	0.0	40
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów	Udział w konsultacjach		Praca własna studenta		RAZEM
	Liczba godzin pracy studenta	40	20.0		65.0		125
Cel przedmiotu	Celem przedmiotu jest nabycie przez studentów wiedzy oraz umiejętności w zakresie programowania i implementacji programów w środowisku Linux/Visual Studio. Student powinien opanować podstawowy repertuar dostępnych w języku C/C++ instrukcji, typów danych, operatorów oraz funkcji. Student powinien nabyć wiedzę na temat struktur, wskaźników oraz innych podstawowych pojęć związanych z programowaniem w języku C/C++.						
Efekty uczenia się przedmiotu	Efekt kierunkowy		Efekt z przedmiotu			Sposób weryfikacji i oceny efektu	
	[K7_W04] zna i rozumie w zaawansowanym stopniu zasady, metody i techniki programowania oraz zasady tworzenia oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo inne elementy lub układy programowalne, specyficznych dla kierunku studiów, a także organizację pracy systemów wykorzystujących komputery lub te urządzenia		Student zna podstawowe zasady programowania języka C/C++.			[SW1] Ocena wiedzy faktograficznej	
	[K7_U04] potrafi wykorzystywać posiadaną wiedzę z zakresu metod i technik programowania oraz dobrać i zastosować właściwe metody i narzędzia programistyczne w tworzeniu oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo elementy lub układy programowalne, charakterystycznych dla danego kierunku studiów, dokonując oceny i krytycznej analizy wykonanego oprogramowania, a także syntezy i twórczej interpretacji prezentowanych za jego pomocą informacji		Student potrafi wykorzystać przekazaną wiedzę (z wykładu), podstawowe techniki języka C/C++ oraz oprogramowanie w systemie Linux/Visual Studio do napisania i kompilacji programu.			[SU1] Ocena realizacji zadania [SU4] Ocena umiejętności korzystania z metod i narzędzi	

Treści przedmiotu	<ol style="list-style-type: none"> 1. Języki programowania, alfabet, syntaktyka i semantyka. Translacja. 2. Klasyfikacja typów. Typy całkowite i zmiennoprzecinkowe. 3. Operatory i wyrażenia arytmetyczne. 4. Wybrane funkcje standardowe. 5. Typ znakowy. Rzutowanie typów. 6. Typ logiczny. Operatory i wyrażenia logiczne. 7. Podstawy obsługi wejścia/wyjścia. 8. Instrukcje warunkowe (if, switch) oraz wyrażenie warunkowe. 9. Instrukcje iteracyjne (for, while, do-while). Iteracje zagnieżdżone. 10. Definiowanie typów. Stałe. Typ wyliczeniowy. 11. Tablice jednowymiarowe i wielowymiarowe. Napisy. 12. Zakres ważności i czas życia zmiennych 13. Funkcje. Zasięg i czas życia zmiennych. Efekt uboczny. 14. Przesyłanie parametrów funkcji. 15. Typ wskaźnikowy. Arytmetyka wskaźników. 16. Wskaźniki w komunikacji między funkcjami. 17. Dynamiczna alokacja pamięci. 18. Struktury 											
Wymagania wstępne i dodatkowe	Brak wymagań											
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 40%;">Sposób oceniania (składowe)</th> <th style="width: 30%;">Próg zaliczeniowy</th> <th style="width: 30%;">Składowa oceny końcowej</th> </tr> </thead> <tbody> <tr> <td>Projekt: poprawność działania programu, algorytmy, struktury danych, czas wykonania oraz uniwersalność programu.</td> <td style="text-align: center;">50.0%</td> <td style="text-align: center;">60.0%</td> </tr> <tr> <td>Egzamin</td> <td style="text-align: center;">50.0%</td> <td style="text-align: center;">40.0%</td> </tr> </tbody> </table>			Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej	Projekt: poprawność działania programu, algorytmy, struktury danych, czas wykonania oraz uniwersalność programu.	50.0%	60.0%	Egzamin	50.0%	40.0%
Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej										
Projekt: poprawność działania programu, algorytmy, struktury danych, czas wykonania oraz uniwersalność programu.	50.0%	60.0%										
Egzamin	50.0%	40.0%										
Zalecana lista lektur	Podstawowa lista lektur	1. KERNIGHAN, Brian W.; RITCHIE, Dennis M. <i>The C programming language</i> , Prentice Hall, 2006										
	Uzupełniająca lista lektur	1. B. Stroustrup, <i>The C++ Programming Language</i> , Addison Wesley Longman, 2000										
	Adresy eZasobów											
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	<p>Co pokaże poniższy fragment kodu, jeśli będzie częścią poprawnego programu?</p> <pre style="margin-left: 20px;">int i; for(i=0;i<3;i++); cout<<i; cout<<i+1;</pre> <p>a) 011223 b) 0124 c) 0123 d) 34 e) 124 f) 45</p>											
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy											