



Karta przedmiotu

Nazwa i kod przedmiotu	Programowanie, PG_00021027						
Kierunek studiów	Matematyka						
Data rozpoczęcia studiów	październik 2021 r.	Rok akademicki realizacji przedmiotu			2021/2022		
Poziom kształcenia	I stopnia - licencjackie	Grupa zajęć			Grupa zajęć obowiązkowych z zakresu kierunku studiów Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne	Sposób realizacji			na uczelni		
Rok studiów	1	Język wykładowy			polski		
Semestr studiów	2	Liczba punktów ECTS			5.0		
Profil kształcenia	ogólnoakademicki	Forma zaliczenia			zaliczenie		
Jednostka prowadząca	Wydział Fizyki Technicznej i Matematyki Stosowanej -> Katedra Rachunku Prawdopodobieństwa i Biomatematyki						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot	dr inż. Magdalena Chmara					
	Prowadzący zajęcia z przedmiotu	dr Adrian Myszkowski mgr inż. Jakub Ciesielski dr inż. Magdalena Chmara					
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	30.0	0.0	30.0	0.0	0.0	60
	W tym liczba godzin zajęć na odległość: 0.0						
	Adres na platformie eNauczanie: <a href="https://enauczanie.pg.edu.pl/moodle/course/view.php?id=21056">https://enauczanie.pg.edu.pl/moodle/course/view.php?id=21056</a> Adresy na platformie eNauczanie: Programowanie lato 2021/2022 - Moodle ID: 21056 <a href="https://enauczanie.pg.edu.pl/moodle/course/view.php?id=21056">https://enauczanie.pg.edu.pl/moodle/course/view.php?id=21056</a>						
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	60		5.0		60.0	125
Cel przedmiotu	Umiejętność pisania prostych algorytmów w wybranym języku programowania; kompilowania, uruchamiania i testowania prostych programów. Opanowanie umiejętności analizowania prostych algorytmów.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[K6_W08] zna podstawy technik obliczeniowych i programowania, wspomagających pracę matematyka i rozumie ich ograniczenia	Student: - rozpoznaje elementy składowe programu i wyjaśnia ich przeznaczenie - wymienia kryteria oceny jakości programów	[SW1] Ocena wiedzy faktograficznej
	[K6_U07] potrafi wykorzystywać narzędzia i metody numeryczne do rozwiązywania wybranych zagadnień rachunku różniczkowego i całkowego, w tym także bazujących na jego zastosowaniach, rozpoznaje problemy, w tym zagadnienia praktyczne, które można rozwiązać algorytmicznie; potrafi dokonać specyfikacji takiego problemu	Student: - projektuje proste algorytmy i ich testy.	[SU4] Ocena umiejętności korzystania z metod i narzędzi
	[K6_W09] zna na poziomie podstawowym co najmniej jeden pakiet oprogramowania, służący do obliczeń symbolicznych	Student: - stosuje narzędzia wytwarzania programu w języku C/ C++, - stosuje internet do znajdowania informacji o języku C/ C++ i programowaniu	[SW1] Ocena wiedzy faktograficznej
	[K6_K03] potrafi pracować zespołowo; rozumie konieczność systematycznej pracy nad wszelkimi projektami, które mają długofalowy charakter, rozumie i docenia znaczenie uczciwości intelektualnej w działaniach własnych i innych osób; postępuje etycznie	Student na każdych laboratoriach pisze kilka programów.	[SK3] Ocena umiejętności organizacji pracy [SK4] Ocena umiejętności komunikacji, w tym poprawności językowej [SK2] Ocena postępów pracy

## Treści przedmiotu

### Wykład:

1. Liczby w języku C/C++. Pamięć operacyjna. Liczby całkowite. Liczby zmiennoprzecinkowe. Wektory i macierze.
2. Iteracja: Procesor i jego działanie. Polecenie warunkowe. Polecenie wyboru. Polecenia iteracyjne. Optymalizacja. Wyszukiwanie i sortowanie wartości liczbowych. Schemat Hornera. Operacje na plikach dyskowych. Złożoność obliczeniowa algorytmów. Cechy dobrego stylu programowania. Testowanie programu.
3. Alfabet i tekst: Kod ASCII i UNICODE. Znaki. Łącuch znaków. Wyszukiwanie i sortowanie łańcuchów znakowych.
4. Procedury i funkcje: Definicja, parametry i zmienne lokalne. Biblioteka funkcji i projekt. Algorytmy rekurencyjne.
5. Struktury danych: Definiowanie struktur danych. Dynamiczny przydział pamięci. Referencja. Elementarne struktury danych.
6. Klasa i obiekt: Definiowanie klas i obiektów oraz ich stosowanie. Konstruktor. Przeciążenie metody i operatora. Funkcja zaprzyjaźniona. Dziedziczenie.

### Laboratorium:

Laboratorium 1: Wprowadzenie do programowania w języku C

Laboratorium 2: Zmienne, Instrukcje warunkowe i wyboru w języku C

Laboratorium 3: Pętla for w języku C

Laboratorium 4: Pętle while i do-while w języku C

Laboratorium 5: Funkcje i rekurencja w języku C

Laboratorium 6: Tablice w języku C++

Laboratorium 7: Łącuchy znaków w języku C++

Laboratorium 8: Obsługa plików w języku C++

Laboratorium 9: Struktury danych w języku C++

Laboratorium 10: Klasy i obiekty w języku C++

Laboratorium 11: Dziedziczenie w C++

Laboratorium 12: GUI

Laboratorium 13: GUI

Laboratorium 14: Obsługa wyjątków, debuggowanie

Laboratorium 15: Podsumowanie semestru

Wymagania wstępne i dodatkowe			
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	Ćwiczenia wykonane na laboratoriach	0.0%	65.0%
	Pięć dłuższych zadań praktycznych	0.0%	15.0%
	Dwa testy z wykładu	50.0%	20.0%
Zalecana lista lektur	Podstawowa lista lektur	1. Brian W. Kernighan, Dennis M. Ritchiem Język ANSI C. Programowanie. Wydanie II, Helion, 2020 2. Jerzy Grębosz, Opus magnum C++11. Programowanie w języku C++. Helion 2020 3. Jerzy Grębosz, Opus magnum C++. Misja w nadprzestrzeni C++14/17. Helion 2020	
	Uzupełniająca lista lektur	Eckel B.: Thinking in C++, Gliwice: Helion, 2004	
	Adresy eZasobów	Podstawowe <a href="https://doi.org/10.1007/978-1-4842-4288-9">https://doi.org/10.1007/978-1-4842-4288-9</a> - Mikael Olsson, Modern C Quick Syntax Reference: A Pocket Guide to the Language, APIs and Library, APRESS 2019, Programowanie lato 2021/2022 - Moodle ID: 21056 <a href="https://enauczanie.pg.edu.pl/moodle/course/view.php?id=21056">https://enauczanie.pg.edu.pl/moodle/course/view.php?id=21056</a>	
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	Zaprojektować algorytm iteracyjny stosujący schemat Hornera i napisać program, w C/C++, realizujący ten algorytm.		
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy		