

SYLABUS

DOTYCZY CYKLU KSZTAŁCENIA 2021-2025

Rok akademicki 2021/2022

1. PODSTAWOWE INFORMACJE O PRZEDMIOCIE

Nazwa przedmiotu	<i>podstawy programowania w języku C</i>
Kod przedmiotu	
Nazwa jednostki prowadzącej kierunek	<i>Kolegium Nauk Przyrodniczych</i>
Nazwa jednostki realizującej przedmiot	<i>Kolegium Nauk Przyrodniczych</i>
Kierunek studiów	<i>informatyka</i>
Poziom studiów	<i>studia I stopnia</i>
Profil	<i>ogólnoakademicki</i>
Forma studiów	<i>stacjonarne</i>
Rok i semestr/y studiów	<i>rok I, semestr 1</i>
Rodzaj przedmiotu	<i>przedmiot kierunkowy</i>
Język wykładowy	<i>język polski</i>
Koordynator	<i>dr inż. Michał Kępski</i>
Imię i nazwisko osoby prowadzącej / osób prowadzących	<i>dr inż. Michał Kępski, mgr inż. Dawid Kosior, mgr inż. Adam Szczur, mgr Roman Hrytsak</i>

1.1. Formy zajęć dydaktycznych, wymiar godzin i punktów ECTS

Semestr (nr)	Wykł.	Ćw.	Konw.	Lab.	Sem.	ZP	Prakt.	Inne (jakie?)	Liczba pkt. ECTS
1	15			30					4

1.2. Sposób realizacji zajęć

zajęcia realizowane częściowo w formie tradycyjnej a częściowo z wykorzystaniem metod i technik kształcenia na odległość

1.3 Forma zaliczenia przedmiotu (z toku)

zaliczenie z oceną

2. WYMAGANIA WSTĘPNE

- Umiejętność logicznego myślenia.
- Znajomość informatyki w zakresie programowania na poziomie szkoły średniej.

3. CELE, EFEKTY UCZENIA SIĘ, TREŚCI PROGRAMOWE I STOSOWANE METODY DYDAKTYCZNE

3.1 Cele przedmiotu

C ₁	Zapoznanie studentów z podstawowymi pojęciami związanymi z programowaniem komputerów.
C ₂	Zapoznanie studentów z podstawowymi zasadami i narzędziami tworzenia programów oraz możliwościami współczesnych języków programowania wysokiego poziomu na przykładzie języka C.
C ₂	Wykształcenie umiejętności tworzenia programów komputerowych przy wykorzystaniu możliwości oferowanych przez język wysokiego poziomu na przykładzie języka C.

3.2 Efekty uczenia się dla przedmiotu

EK (efekt uczenia się)	Treść efektu uczenia się zdefiniowanego dla przedmiotu Student:	Odniesienie do efektów kierunkowych ¹
EK_01	Zna podstawowe pojęcia związane z programowaniem komputerów.	K_W03, K_W04
EK_02	Zna podstawowe struktury danych oraz wybrane konstrukcje programistyczne dostępne w języku C.	K_W07
EK_03	Potrafi wykorzystać odpowiednie struktury danych oraz konstrukcje programistyczne dostępne w języku C przy tworzeniu programów rozwiązujących zadane problemy.	K_U10 - K_U12
EK_04	Potrafi tworzyć programy z wykorzystaniem standardowych bibliotek programistycznych dostępnych w języku C.	K_U10 - K_U12

3.3 Treści programowe

A. Problematyka wykładu

1. Podstawowe pojęcia związane z programowaniem komputerów. Programowanie strukturalne. Ogólne cechy języka C.
2. Struktura programu w języku C. Zmienne i stałe. Typy danych, literały, operatory i wyrażenia. Operacje wejścia i wyjścia.
3. Instrukcje sterujące: <i>if</i> , <i>if-else</i> , <i>switch</i> , <i>while</i> , <i>do-while</i> , <i>for</i> . Zagnieżdżanie instrukcji. Instrukcje przerwań <i>break</i> i <i>continue</i> .
4. Wskaźniki. Tablice jedno i wielowymiarowe. Łańcuchy znaków.
5. Funkcje. Przekazywanie parametrów do funkcji. Zmienne globalne i lokalne. Rekurencja. Wybrane funkcje biblioteczne.
6. Operacje na plikach. Biblioteka standardowa.
7. Struktury. Unie. Dynamiczny przydział pamięci.

B. Problematyka ćwiczeń laboratoryjnych

1. Środowisko programistyczne – konfiguracja. Tworzenie prostych programów w języku C.
2. Tworzenie programów w języku C wykorzystujących instrukcje sterujące: <i>if</i> , <i>if-else</i> , <i>switch</i> , <i>while</i> , <i>do-while</i> , <i>for</i> , zagnieżdżanie instrukcji, instrukcje przerwań <i>break</i> i <i>continue</i> .
3. Tworzenie programów w języku C wykorzystujących wskaźniki, tablice jedno i wielowymiarowe, łańcuchy znaków.

¹ W przypadku ścieżki kształcenia prowadzącej do uzyskania kwalifikacji nauczycielskich uwzględnić również efekty uczenia się ze standardów kształcenia przygotowującego do wykonywania zawodu nauczyciela.

4. Tworzenie programów w języku C wykorzystujących funkcje.
5. Tworzenie programów w języku C wykorzystujących operacje na plikach.
6. Tworzenie programów w języku C wykorzystujących struktury i unie.

3.4 Metody dydaktyczne

Wykład: wykład z prezentacją multimedialną.

Laboratorium: wykonywanie praktycznych zadań programistycznych – praca przy komputerze.

4. METODY I KRYTERIA OCENY

4.1 Sposoby weryfikacji efektów uczenia się

Symbol efektu	Metody oceny efektów uczenia się (np.: kolokwium, egzamin ustny, egzamin pisemny, projekt, sprawozdanie, obserwacja w trakcie zajęć)	Forma zajęć dydaktycznych (w, ćw, ...)
EK_01	Pisemne zaliczenie wykładu	Wykład
EK_02	Pisemne zaliczenie wykładu	Wykład
EK_03	Kolokwium	Laboratorium
EK_04	Kolokwium	Laboratorium

4.2 Warunki zaliczenia przedmiotu (kryteria oceniania)

Efekt EK_01:

Zaliczenie: Student potrafi omówić co najmniej 50% pojęć związanych z programowaniem komputerów.

Efekt EK_02:

Zaliczenie: Student potrafi poprawnie odpowiedzieć na co najmniej 50% pytań związanych ze strukturami danych oraz wybranymi konstrukcjami programistycznymi dostępnymi w języku C.

Laboratorium – kolokwium praktyczne przy komputerze

Efekt EK_03:

DST: Student potrafi poprawnie przygotować programy rzędu 20-50 linii kodu, rozwiązujące proste problemy, wykorzystując do tego celu niektóre z poznanych struktur danych oraz konstrukcje programistyczne dostępne w języku C.

DB: Student potrafi poprawnie przygotować programy rzędu 20-50 linii kodu, rozwiązujące średniozaawansowane problemy, wykorzystując do tego celu wszystkie poznane struktury danych oraz konstrukcje programistyczne dostępne w języku C.

BDB: Student potrafi poprawnie przygotować programy rzędu 50-100 linii kodu rozwiązujące średniozaawansowane problemy, wykorzystując do tego celu wszystkie poznane struktury danych oraz konstrukcje programistyczne dostępne w języku C.

Efekt EK_04:

DST: Student potrafi poprawnie przygotować programy rzędu 20-50 linii kodu, rozwiązujące proste problemy wykorzystując do tego celu możliwości standardowych bibliotek programistycznych dostępnych w języku C używanych na zajęciach.

DB: Student potrafi poprawnie przygotować programy rzędu 20-50 linii kodu, rozwiązujące średniozaawansowane problemy, wykorzystując możliwości standardowych bibliotek programistycznych dostępnych w języku C używanych na zajęciach.

BDB: Student potrafi poprawnie przygotować programy rzędu 50-100 linii kodu, rozwiązujące średniozaawansowane problemy, wykorzystując możliwości standardowych bibliotek programistycznych dostępnych w języku C, także takich, które student poznawał samodzielnie.

5. CAŁKOWITY NAKŁAD PRACY STUDENTA POTRZEBNY DO OSIĄGNIĘCIA ZAŁOŻONYCH EFEKTÓW W GODZINACH ORAZ PUNKTACH ECTS

Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności
Godziny kontaktowe wynikające z harmonogramu studiów	45
Inne z udziałem nauczyciela akademickiego (udział w konsultacjach, egzaminie)	2
Godziny niekontaktowe – praca własna studenta (przygotowanie do zajęć, egzaminu, napisanie referatu itp.)	58
SUMA GODZIN	105
SUMARYCZNA LICZBA PUNKTÓW ECTS	4

6. PRAKTYKI ZAWODOWE W RAMACH PRZEDMIOTU

Nie dotyczy

7. LITERATURA

Literatura podstawowa:

1. Kernighan B.W., Ritchie D.M.: Język ANSI C: Programowanie. wyd. Helion, Gliwice, 2010.
2. Prata S.: Język C. Wyd. Helion, Gliwice, 2006.
3. Tondo C.L., Gimpel S.E.: Język ANSI C: Programowanie, ćwiczenia. wyd. Helion, Gliwice, 2010.

Literatura uzupełniająca:

1. Shaw Z.A: Programowanie w C: sprytne podejście do trudnych zagadnień, których wolałbyś unikać. Helion, Gliwice, 2014.
2. Reese R.: Wskaźniki w języku C: Przewodnik. Wyd. Helion, Gliwice, 2014.

Akceptacja Kierownika Jednostki lub osoby upoważnionej