

SYLABUS

DOTYCZY CYKLU KSZTAŁCENIA 2022/2023-2025/2026

(skrajne daty)

Rok akademicki 2022/2023

1. PODSTAWOWE INFORMACJE O PRZEDMIOCIE

Nazwa przedmiotu	Podstawy programowania w języku C
Kod przedmiotu*	
Nazwa jednostki prowadzącej kierunek	<i>Kolegium Nauk Przyrodniczych</i>
Nazwa jednostki realizującej przedmiot	<i>Kolegium Nauk Przyrodniczych, Instytut Informatyki</i>
Kierunek studiów	<i>Informatyka i ekonometria</i>
Poziom studiów	<i>studia I stopnia</i>
Profil	<i>praktyczny</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>
Koordinator	<i>dr inż. Michał Kępski</i>
Imię i nazwisko osoby prowadzącej / osób prowadzących	<i>dr inż. Michał Kępski, mgr inż. Ewa Żestawska</i>

* -opcjonalnie, zgodnie z ustaleniami w Jednostce

1.1. Formy zajęć dydaktycznych, wymiar godzin i punktów ECTS

Semestr (nr)	Wykt.	Ćw.	Konw.	Lab.	Sem.	ZP	Prakt.	Inne (jakie?)	Liczba pkt. ECTS
1	15			15					2

1.2. Sposób realizacji zajęć

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

1.3 Forma zaliczenia przedmiotu (z toku) (egzamin, zaliczenie z oceną, zaliczenie bez oceny)

ZALICZENIE Z OCENĄ

2. WYMAGANIA WSTĘPNE

- | |
|--|
| <ul style="list-style-type: none"> - Umiejętność logicznego myślenia. - Znajomość matematyki na poziomie szkoły ponadgimnazjalnej. |
|--|

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	Odniesienie do efektów kierunkowych ¹
EK_01	Zna podstawowe pojęcia związane z programowaniem komputerów.	K_W03
EK_02	Zna podstawowe struktury danych oraz wybrane konstrukcje programistyczne dostępne w języku C.	K_W03
EK_03	Potrafi wykorzystać odpowiednie struktury danych oraz konstrukcje programistyczne dostępne w języku C przy tworzeniu programów rozwiązujących zadane problemy.	K_U01, K_U06
EK_04	Potrafi tworzyć programy z wykorzystaniem standardowych bibliotek programistycznych dostępnych w języku C.	K_U01, K_U06

3.3 Treści programowe

A. Problematyka wykładu

Treści merytoryczne
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: if, if-else, switch, while, do-while, for. Zagnieżdżanie instrukcji. Instrukcje przerwań break i continue.
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.
7. Struktury. Dynamiczny przydział pamięci.

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

B. Problematyka ćwiczeń laboratoryjnych

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

Laboratorium: praktyczne zadania programistyczne 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)

<p><u>Efekt EK_01:</u></p> <p>Zaliczenie: Student potrafi omówić co najmniej 50% pojęć związanych z programowaniem komputerów.</p> <p><u>Efekt EK_02:</u></p> <p>Zaliczenie: Student potrafi poprawnie odpowiedzieć na co najmniej 50% pytań związanych ze strukturami danych oraz wybranymi konstrukcjami programistycznymi dostępnymi w języku C.</p> <p>Laboratorium – kolokwium praktyczne przy komputerze</p> <p><u>Efekt EK_03:</u></p> <p>DST: Student potrafi poprawnie stworzyć program rozwiązujący proste problemy wykorzystując odpowiednie struktury danych oraz konstrukcje programistyczne dostępne w języku C.</p>

DB: Student potrafi poprawnie stworzyć programy rozwiązujące średnio-zaawansowane problemy wykorzystując odpowiednie struktury danych oraz konstrukcje programistyczne dostępne w języku C.

BDB: Student potrafi poprawnie stworzyć programy rozwiązujące zaawansowane problemy wykorzystując odpowiednie struktury danych oraz konstrukcje programistyczne dostępne w języku C.

Efekt EK 04:

DST: Student potrafi poprawnie stworzyć program rozwiązujący proste problemy wykorzystując możliwości standardowych bibliotek programistycznych dostępnych w języku C.

DB: Student potrafi poprawnie stworzyć programy rozwiązujące średnio-zaawansowane problemy wykorzystując możliwości standardowych bibliotek programistycznych dostępnych w języku C.

BDB: Student potrafi poprawnie stworzyć programy rozwiązujące zaawansowane problemy wykorzystując możliwości standardowych bibliotek programistycznych dostępnych w języku C.

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	30
Inne z udziałem nauczyciela akademickiego (udział w konsultacjach, egzaminie)	
Godziny niekontaktowe – praca własna studenta (przygotowanie do zajęć, egzaminu, napisanie referatu itp.)	25
SUMA GODZIN	55
SUMARYCZNA LICZBA PUNKTÓW ECTS	2

* Należy uwzględnić, że 1 pkt ECTS odpowiada 25-30 godzin całkowitego nakładu pracy studenta.

6. PRAKTYKI ZAWODOWE W RAMACH PRZEDMIOTU

wymiar godzinowy	-
zasady i formy odbywania praktyk	-

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 : sprytnie podejście do trudnych zagadnień, których wołałbyś unikać. Helion, Gliwice, 2014.
2. Reese R.: Wskaźniki w języku C: Przewodnik. Wyd. Helion, Gliwice, 2014.

Akceptacja Kierownika Jednostki lub osoby upoważnionej