

SYLABUS

DOTYCZY CYKLU KSZTAŁCENIA 2022/2023 – 2025/2026

(skrajne daty)

Rok akademicki 2022/2023

1. PODSTAWOWE INFORMACJE O PRZEDMIOCIE

Nazwa przedmiotu	Podstawy programowania
Kod przedmiotu*	
Nazwa jednostki prowadzącej kierunek	Kolegium Nauk Przyrodniczych
Nazwa jednostki realizującej przedmiot	Kolegium Nauk Przyrodniczych
Kierunek studiów	Mechatronika
Poziom studiów	Studia I-go stopnia
Profil	Praktyczny
Forma studiów	Studia niestacjonarne
Rok i semestr/y studiów	Rok I, semestr 1
Rodzaj przedmiotu	Przedmiot podstawowy
Język wykładowy	Polski
Koordinator	dr hab. inż. Krzysztof Szemela, prof. UR
Imię i nazwisko osoby prowadzącej / osób prowadzących	dr hab. inż. Krzysztof Szemela, prof. UR

* -opcjonalnie, zgodnie z ustaleniami w Jednostce

1.1. Formy zajęć dydaktycznych, wymiar godzin i punktów ECTS

Semestr (nr)	Wykł.	Ćw.	Konw.	Lab.	Sem.	ZP	Prakt.	Inne (jakie?)	Liczba pkt. ECTS
2	9	18		18					4

1.2. Sposób realizacji zajęć

zajęcia w formie tradycyjnej

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

Wykład - zaliczenie bez oceny, ćwiczenia - zaliczenie z oceną, laboratorium - zaliczenie z oceną

2. WYMAGANIA WSTĘPNE

Znajomość zagadnień z podstaw informatyki i logiki matematycznej.

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

3.1 Cele przedmiotu

C1	Zapoznanie ze składnią języka C
C2	Zapoznanie ze środowiskiem programistycznym dedykowanym dla języka C
C3	Zapoznanie z podstawowymi algorytmami
C4	Wdrożenie do rozwiązywania podstawowych problemu programistycznych w oparciu o język 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 i rozumie zagadnienia z algorytmiki, języków programowania, baz danych i inżynierii oprogramowania, a także z architektury komputerów i systemów operacyjnych w zakresie niezbędnym do stosowania w systemach mechatronicznych wraz z wybranymi zagadnieniami z zakresu budowy oraz zarządzania sieciami komputerowymi	K_Wo9
EK_02	potrafi pozyskiwać informacje z literatury, baz danych i innych źródeł, integrować uzyskane informacje, dokonywać ich interpretacji, a także wyciągać wnioski oraz formułować i uzasadniać opinie	K_Uo1
EK_03	potrafi planować i przeprowadzać eksperymenty, w tym pomiary i symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski, wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne, a także dokonać krytycznej analizy sposobu funkcjonowania i ocenić istniejące rozwiązania techniczne, w szczególności urządzenia, obiekty, systemy i procesy	K_Uo4
EK_04	potrafi zgodnie z zadaną specyfikacją, uwzględniającą aspekty pozatechniczne, zaprojektować złożone urządzenie, obiekt, system lub proces oraz zrealizować ten projekt, co najmniej w części, używając właściwych metod, technik i narzędzi oraz opracować i przedstawić dokumentację dotyczącą realizacji zadania inżynierskiego	K_Uo7
EK_05	potrafi posługiwać się językiem obcym na poziomie B2	K_U16
EK_06	potrafi zaplanować proces własnego uczenia się rozumiejąc potrzebę oraz możliwości ciągłego dokształcania się	K_U19
EK_07	jest gotów do krytycznej oceny własnej wiedzy oraz	K_Ko1

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

	wynikających z niej aspektów i skutków działalności inżyniera – np. wpływu na środowisko i związanej z tym odpowiedzialności za podejmowane decyzje	
--	---	--

3.3 Treści programowe

A. Problematyka wykładu

Treści merytoryczne
1. Podstawowe elementy języka ANSI C. Programowanie proceduralne – ogólne założenia. Budowa programu, pliki nagłówkowe, operacje wejścia i wyjścia, semantyka zmiennej, zapis wyrażań arytmetycznych. (1 godz.)
2. Przegląd konstrukcji języka C. Typy danych i ich opis. Specyfikatory i modyfikatory. Definiowanie typów. Instrukcje proste: przypisania, skoku, pusta. Przykłady. (1 godz.)
3. Instrukcje strukturalne. Instrukcja warunkowa if...else pojedyncza i zagnieżdżona. Instrukcja wyboru switch. Przykłady. (2 godz.)
4. Pętle. Organizowanie obliczeń cyklicznych o znanej ilości powtórzeń - pętla for dla zmiennych prostych. Warunek zakończenia działania pętli – instrukcje while i do while. Procedury continue i break. Obliczenia skończone i nieskończone. Przykłady algorytmów. (2 godz.)
5. Funkcje. Definiowanie i wywoływanie funkcji. Zwracanie wartości, typ void. Przekazywanie parametrów, pojęcie wskaźnika. Parametry formalne i aktualne. Zakres widzialności. Rekurencja. Przykłady. (2 godz.)
6. Typy strukturalne: tablice jednowymiarowe i wielowymiarowe. Deklaracja, wprowadzanie i wyprowadzanie danych, przykłady zastosowań. (1 godz.)

B. Problematyka ćwiczeń audytoryjnych i laboratoryjnych

Treści merytoryczne
1. Zmienne i typy danych. Instrukcja przypisania. Operatory i ich priorytety. Tłumaczenie wyrażań arytmetycznych. (1 godz.)
2. Podstawowe funkcje matematyczne z biblioteki <math.h>. Biblioteka <stdio.h> – obsługa wejścia/wyjścia. Kody formatujące i znaki specjalne dla funkcji wejścia/wyjścia. (2 godz.)
3. Instrukcje sterujące if oraz switch. (2 godz.)
4. Organizowanie obliczeń cyklicznych o znanej ilości powtórzeń - pętla for dla zmiennych prostych. (2 godz.)
5. Warunek zakończenia obliczeń pętli - instrukcje while i do.. while . Algorytmy wyboru ekstremum, sumowanie szeregu z zadaną dokładnością, tablicowanie funkcji, obliczenia z krokiem, itp. (2 godz.)
6. Definicje i wywołanie funkcji. Typ void . Przekazywanie parametrów przez wartość i przez wskaźnik. Zmienne globalne/lokalne. (2 godz.)
7. Zmienne indeksowane. Obsługa ciągów znakowych. Wskaźniki – podstawy. Znakowe i łańcuchowe wejście/wyjście. (2 godz.)
8. Operacje na wskaźnikach. (2 godz.)
9. Tablice jedno i dwuwymiarowe – Algorytmy zliczania, porównywania, porządkowania w odniesieniu do zmiennych tablicowych. (2 godz.)
10. Kolokwium (1 godz.)

3.4 Metody dydaktyczne

Wykład - prezentacja multimedialna,

Ćwiczenia – rozwiązywanie zadań, analiza przykładów, dyskusja,

Laboratorium – rozwiązywanie zadań, praca w grupach, analiza przykładów, dyskusja.

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	Kolokwium	lab., ćw.
EK_02	Kolokwium	lab., ćw.
EK_03	Kolokwium, obserwacja w trakcie zajęć	lab., ćw.
EK_04	Kolokwium, obserwacja w trakcie zajęć	lab., ćw.
EK_05	Kolokwium	lab., ćw.
EK_06	Kolokwium	lab., ćw.
EK_07	Kolokwium	lab., ćw.

4.2 Warunki zaliczenia przedmiotu (kryteria oceniania)

<p>Wykład - warunkiem zaliczenia jest zdobycie zaliczenia z ćwiczeń i laboratorium, Ćwiczenia - zaliczenie kolokwium, Laboratorium - zaliczenie kolokwium praktycznego.</p> <p>Dodatkowym warunkiem zaliczenia przedmiotu jest co najwyżej jedna nieobecność nieusprawiedliwiona na ćwiczeniach audytoryjnych i laboratoriach.</p> <p>Ponadwymiarowe nieobecności muszą być nadrobione wykonaniem dodatkowych zadań wskazanych przez prowadzącego zajęcia.</p>
--

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

Forma aktywności	Średnia liczba godzinna zrealizowanie aktywności
Godziny kontaktowe wynikające z harmonogramu studiów	45
Inne z udziałem nauczyciela akademickiego (udział w konsultacjach, egzaminie)	5
Godziny niekontaktowe – praca własna studenta (przygotowanie do zajęć, egzaminu, napisanie referatu itp.)	60
SUMA GODZIN	110
SUMARYCZNA LICZBA PUNKTÓW ECTS	4

* 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. S. Prata, „Język C. Szkoła programowania. Wydanie V”, Helion 2008
2. B. W. Kernighan, D. M. Ritchie, „Język ANSI C : programowanie”, Helion, 2010
3. K. N. King, „Język C : nowoczesne programowanie”, Helion, 2011
4. C. L. Tondo, S. E. Gimpel, „Język ANSI C : programowanie, ćwiczenia”, Helion, 2010
5. <http://upload.wikimedia.org/wikibooks/pl/6/6a/C.pdf> - książka dostępna na portalu wikibooks
6. http://www.acm.uiuc.edu/webmonkeys/book/c_guide/

Literatura uzupełniająca:

7. B. Borowik, W. Borowik, B. Borowik, „Meandry języka C/C++”, Mikom 2006
8. P. Stańczyk, „Algorytmika praktyczna”, PWM, 2009
9. K. Wojtuszkiewicz, „Programowanie strukturalne i obiektowe T. I i T.II”, PWM, 2009
10. <http://4programmers.net/>
11. <http://computer.howstuffworks.com/c1.htm>
12. „Programowanie w C.pdf” – Wikibooks
<http://www.cplusplus.com/reference/clibrary/>

Akceptacja Kierownika Jednostki lub osoby upoważnionej