SYLABUS

DOTYCZY CYKLU KSZTAŁCENIA 2026/2027 - 2029/2030

Rok akademicki 2026/2027

1. PODSTAWOWE INFORMACJE O PRZEDMIOCIE

Nazwa przedmiotu	*programowanie w języku C*
Kod przedmiotu*	
Nazwa jednostki prowadzącej kierunek	*Instytut Informatyki, Wydział Nauk Ścisłych i Technicznych*
Nazwa jednostki realizującej przedmiot	*Instytut Informatyki, Wydział Nauk Ścisłych i Technicznych*
Kierunek studiów	*sztuczna inteligencja*
Poziom studiów	*studia I stopnia*
Profil	*ogólnoakademicki*
Forma studiów	*stacjonarne*
Rok i semestr/y studiów	*rok I, semestr 1*
Rodzaj przedmiotu	*przedmiot kierunkowy*
Język wykładowy	*polski*
Koordynator	*dr inż. Michał Kępski*
Imię i nazwisko osoby prowadzącej / osób prowadzących	*dr inż. Michał Kępski*

***** - *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**
1	15			30					3

1.2. Sposób realizacji zajęć

zajęcia w formie tradycyjnej

1.3 Forma zaliczenia przedmiotu (z toku)

wykład – zaliczenie bez oceny, laboratoria – zaliczenie z oceną

2.WYMAGANIA WSTĘPNE

Umiejętność programowania na poziomie szkoły średniej.

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

3.1 Cele przedmiotu

C1	Usystematyzowanie wiadomości ze szkoły średniej na temat programowania komputerów.
C2	Zapoznanie studentów z podstawowymi zasadami i narzędziami tworzenia programów oraz możliwościami języków programowania na przykładzie języka C.
C3	Wykształcenie / poprawienie umiejętności programowania w języku 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 i konstrukcje programistyczne języka C.	K_W03
EK_02	Potrafi wykorzystać odpowiednie typy danych oraz konstrukcje programistyczne języka C przy tworzeniu programów rozwiązujących zadane problemy.	K_U03
EK_03	Potrafi programować wykorzystując bibliotekę standardową języka C. Potrafi programować wykorzystując dynamiczną alokację pamięci, wskaźniki, struktury oraz unie.	K_U03

3.3 Treści programowe

A. Problematyka wykładu

1. Podstawowe pojęcia związane z programowaniem komputerów. Podział języków programowania. System komputerowy z punktu widzenia programisty. Ogólne cechy języka C. Pierwszy program. Kompilowanie kodu źródłowego.
2. Struktura programu. Zmienne i stałe. Typy danych, literały, operatory i wyrażenia. Instrukcje sterujące. Zagnieżdżanie instrukcji. Instrukcje przerwań `break` i `continue`.
3. Funkcje. Przekazywanie parametrów do funkcji. Zmienne globalne i lokalne. Rekurencja. Operacje wejścia i wyjścia.
4. Wskaźniki. Tablice jedno- i wielowymiarowe. Tablice znaków. Alokacja pamięci.
5. Struktury i unie.
6. Wybrane elementy biblioteki standardowej. Operacje na plikach.
7. Pamieć wirtualna. Sterta i stos.

B. Problematyka laboratoriów

1. Środowisko programistyczne – konfiguracja. Tworzenie prostych programów w języku C. Funkcje `scanf()` i `printf()` do obsługi wejścia/wyjścia standardowego.
2. Tworzenie programów z instrukcjami sterującymi: `if`, `if-else`, `switch`. Zagnieżdżanie instrukcji.
3. Tworzenie programów z pętlą: `while`, `do-while`, `for`. Instrukcje `break` i `continue`.
4. Definiowanie własnych funkcji. Parametry funkcji i wartości zwracane.

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

5. Tablice w języku C. Łańcuchy znaków, tablice wielowymiarowe.
6. Rekurencja w języku C.
7. Wskaźniki w języku C. Arytmetyka wskaźników.
8. Operacje na plikach. Odczyt, zapis, dostęp swobodny. Strumienie.
9. Struktury i unie.
10. Alokacja pamięci.

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	kolokwium	w
EK_02	kolokwium	lab
EK_03	kolokwium	lab

4.2 Warunki zaliczenia przedmiotu (kryteria oceniania)

WYKŁAD – TEST

EFEKT EK_01:

Zaliczenie: student potrafi poprawnie odpowiedzieć na co najmniej 50% pytań związanych pojęciami związanymi z programowaniem komputerów i konstrukcjami programistycznymi dostępnymi w języku C.

LABORATORIUM – KOLOKWIUM PRAKTYCZNE PRZY KOMPUTERZE

EFEKT EK_02:

DST: Student potrafi poprawnie przygotować programy rozwiązujące proste problemy, wykorzystując do tego celu niektóre z poznanych typów danych oraz konstrukcje programistyczne dostępne w języku C.

DB: Student potrafi poprawnie przygotować programy rozwiązujące średniozaawansowane problemy, wykorzystując do tego celu niektóre z poznanych typów danych oraz konstrukcje programistyczne dostępne w języku C.

BDB: Student potrafi poprawnie przygotować programy rozwiązujące zaawansowane problemy, wykorzystując do tego celu niektóre z poznanych typów danych oraz konstrukcje programistyczne dostępne w języku C.

EFEKT EK_03:

DST: Student potrafi poprawnie przygotować programy, rozwiązujące proste problemy wykorzystując do tego celu możliwości standardowych bibliotek programistycznych dostępnych w języku C używanych na zajęciach. Potrafi poprawnie przygotować programy operujące na wskaźnikach, prostych strukturach i uniach.

DB: Student potrafi poprawnie przygotować programy, rozwiązujące średniozaawansowane problemy, wykorzystując możliwości standardowych bibliotek programistycznych dostępnych w języku C używanych na zajęciach. Potrafi poprawnie przygotować programy operujące na wskaźnikach (także wskaźnikach do funkcji), strukturach i uniach. Definiuje własne struktury i unie.

BDB: Student potrafi poprawnie przygotować programy, rozwiązujące zaawansowane problemy, wykorzystując możliwości standardowych bibliotek programistycznych dostępnych w języku C, także takich, które student poznawał samodzielnie. Potrafi poprawnie przygotować programy operujące na wskaźnikach (także wskaźnikach do funkcji), złożonych strukturach i uniach i ich tablicach. Definiuje własne struktury i unie, w tym także takie, które wymagają dynamicznego przydziału pamięci.

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

Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności
Godziny 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.)	43
SUMA GODZIN	90
SUMARYCZNA LICZBA PUNKTÓW ECTS	**3**

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