



Programowanie w języku C Sylabus zajęć

Informacje podstawowe

Kierunek studiów Informatyka	Cykl dydaktyczny 2023/24
Specjalność -	Kod zajęć 06INFS.41S.01003.23
Jednostka organizacyjna Wydział Matematyki i Informatyki	Języki wykładowe polski
Poziom studiów studia drugiego stopnia poinżynierskie	Obligatoryjność Fakultatywny
Forma studiów studia stacjonarne	Blok zajęciowy Przedmioty specjalnościowe
Profil studiów profil ogólnoakademicki	
Koordynator zajęć	Bartłomiej Przybylski
Prowadzący zajęcia	Bartłomiej Przybylski
Okres Semestr 1	Forma zajęć / liczba godzin / forma zaliczenia • Laboratorium: 30, Zaliczenie z oceną
	Liczba punktów ECTS 3

Cele kształcenia dla zajęć

Kod	Cel
C1	Uwypuklenie znaczenia języka ANSI C w budowaniu aplikacji dla współczesnych systemów operacyjnych.
C2	Wykształcenie umiejętności sprawnego posługiwania się językiem ANSI C, w tym w zakresie komunikacji z systemem operacyjnym.
C3	Doskonalenie umiejętności implementacji wybranych algorytmów w języku programowania.
C4	Doskonalenie warsztatu programisty.

Wymagania wstępne

Znajomość dowolnego języka programowania oraz podstawowych zagadnień z zakresu zarządzania zasobami w systemach operacyjnych.

Efekty uczenia się dla zajęć

Kod	Efekty uczenia się dla zajęć w zakresie	Efekty uczenia się dla kierunku	Metody weryfikacji osiągnięcia efektów uczenia się dla zajęć
Wiedzy - Student/ka:			
W1	zna i rozumie koncepcje języka ANSI C oraz jego ograniczenia.	INF_K4_W04, INF_K4_W05	Projekt, Zadania wykonywane podczas zajęć oraz w domu
W2	zna i rozumie rolę, możliwości i ograniczenia biblioteki standardowej.	INF_K4_W04, INF_K4_W05	Projekt, Zadania wykonywane podczas zajęć oraz w domu
W3	zna i rozumie zasady zarządzania zasobami w systemach uniksopodobnych.	INF_K4_W04	Projekt, Zadania wykonywane podczas zajęć oraz w domu
W4	zna i rozumie problemy współbieżności i synchronizacji.	INF_K4_W01, INF_K4_W02	Projekt, Zadania wykonywane podczas zajęć oraz w domu
Umiejętności - Student/ka:			
U1	potrafi posługiwać się językiem ANSI C.	INF_K4_U03	Projekt, Zadania wykonywane podczas zajęć oraz w domu
U2	potrafi korzystać z biblioteki standardowej oraz funkcji systemowych.	INF_K4_U03	Projekt, Zadania wykonywane podczas zajęć oraz w domu
U3	potrafi posługiwać się kompilatorem oraz wybranymi narzędziami wspomagającymi proces kompilacji.	INF_K4_U03	Projekt, Zadania wykonywane podczas zajęć oraz w domu
U4	potrafi implementować proste programy wielowątkowe.	INF_K4_U02, INF_K4_U03, INF_K4_U04	Projekt, Zadania wykonywane podczas zajęć oraz w domu
U5	potrafi utworzyć bibliotekę statyczną i dynamiczną.	INF_K4_U02, INF_K4_U03, INF_K4_U04	Projekt, Zadania wykonywane podczas zajęć oraz w domu
U6	potrafi pogłębiać swoją wiedzę w oparciu o dokumentację języka i wykorzystywane narzędzia.	INF_K4_U11	Projekt, Zadania wykonywane podczas zajęć oraz w domu
U7	potrafi samodzielnie zgłębiać wiedzę z obszarów programowania i systemów operacyjnych.	INF_K4_U12	Projekt, Zadania wykonywane podczas zajęć oraz w domu

Treści programowe dla zajęć

Lp.	Treści programowe dla zajęć	Efekty uczenia się dla zajęć	Formy zajęć
1.	Podstawowe koncepcje języka ANSI C. Zmienne i funkcje, struktury i unie, wskaźniki i tablice, operatory, instrukcje warunkowe, pętle, podstawowe operacje wejścia/wyjścia. Elementy procesu kompilacji. Narzędzia gcc i clang.	W1, U1, U3, U6	Laboratorium
2.	Standardowe wejście i wyjście. Argumenty przekazywane podczas wywoływania programu.	W1, U1	Laboratorium
3.	Biblioteka standardowa i matematyczna. Samodzielna implementacja wybranych funkcji z biblioteki standardowej.	W2, U2, U6	Laboratorium
4.	Funkcje systemowe. Operacje na plikach. Samodzielna implementacja wybranych narzędzi (np. cat i cp) z wykorzystaniem funkcji systemowych.	W3, U2, U6, U7	Laboratorium
5.	Struktura procesu. Stos i sarta. Zarządzanie pamięcią i zaawansowane operacje na wskaźnikach. Samodzielna implementacja podstawowych struktur danych (stos, lista z dowiązaniem, drzewo wyszukiwań binarnych).	W3, U2, U3, U6, U7	Laboratorium
6.	Funkcje systemowe. Tworzenie procesów i zarządzanie współdzielonymi zasobami.	W3, U2	Laboratorium
7.	Współbieżność i synchronizacja. Obsługa współbieżności przez jądro. Wątki POSIX i synchronizacja wątków.	W3, W4, U4	Laboratorium
8.	Tworzenie bibliotek statycznych i dynamicznych.	W3, U5, U6	Laboratorium
9.	Integracja kodu w języku C z Pythonem.	W3, U4, U6	Laboratorium

Informacje dodatkowe

Forma zajęć	Metody i formy prowadzenia zajęć
Laboratorium	Dyskusja, Praca z tekstem, Metoda analizy przypadków, Uczenie problemowe (Problem-based learning), Rozwiązywanie zadań (np.: obliczeniowych, artystycznych, praktycznych), Metoda laboratoryjna, Metoda projektu

Forma zajęć	Warunki zaliczenia zajęć
Laboratorium	Końcowa ocena składa się z następujących elementów: 1. projekt – 50%, 2. zadania wykonywane podczas zajęć oraz w domu – 50%. Skala ocen: 1. bardzo dobry (bdb; 5,0) – od 90% punktów, 2. dobry plus (db plus; 4,5) – od 80% punktów, 3. dobry (db; 4,0) – od 70% punktów, 4. dostateczny plus (dst plus; 3,5) – od 60% punktów, 5. dostateczny (dst; 3,0) – od 50% punktów, 6. niedostateczny (ndst; 2,0) – poniżej 50% punktów.

Literatura

Obowiązkowa

1. B. W. Kernighan, D. Ritchie, "Język ANSI C", Wydawnictwa Naukowo-Techniczne (2000)
2. C. L. Tondo, S. E. Gimpel, "Język ANSI C. Ćwiczenia i rozwiązania", Wydawnictwa Naukowo-Techniczne (2003)
3. A. Kamran, "Ekstremalny kod w języku C", Wydawnictwo Helion (2021)

Dodatkowa

1. T. H. Cormen i in., "Wprowadzenie do algorytmów", Wydawnictwo Naukowe PWN (2012)

Nakład pracy studenta i punkty ECTS

Rodzaje zajęć studenta	Średnia liczba godzin* przeznaczonych na zrealizowane rodzaje zajęć
Laboratorium	30
Przygotowanie do zajęć	10
Przygotowanie projektu	10
Inne	25
Łączny nakład pracy studenta	Liczba godzin 75
Liczba punktów ECTS	ECTS 3

* godzina (lekcyjna) oznacza 45 minut

Efekty uczenia się dla kierunku

Kod	Treść
INF_K4_U02	Absolwent/ka potrafi adaptować istniejące oraz tworzyć nowe metody informatyczne do rozwiązywania nieszablonowych problemów praktycznych i teoretycznych
INF_K4_U03	Absolwent/ka potrafi stosować zaawansowane metody budowy oprogramowania, rozstrzyga o ich przydatności, w tym podejmuje decyzje dotyczące wyboru technik prowadzących do otrzymania oprogramowania wysokiej jakości
INF_K4_U04	Absolwent/ka potrafi projektować i implementować systemy informatyczne o różnej złożoności i różnych architekturach
INF_K4_U11	Absolwent/ka potrafi pozyskiwać informacje z literatury, baz wiedzy, Internetu oraz innych wiarygodnych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie
INF_K4_U12	Absolwent/ka potrafi samodzielnie pogłębiać i aktualizować wiedzę i umiejętności z zakresu informatyki oraz określać kierunki dalszego rozwoju zawodowego
INF_K4_W01	Absolwent/ka zna i rozumie w pogłębionym stopniu pojęcia z działów matematyki niezbędne do rozwiązywania zaawansowanych problemów w informatyce
INF_K4_W02	Absolwent/ka zna i rozumie współczesny stan badań i tendencje rozwojowe w wiodących obszarach informatyki
INF_K4_W04	Absolwent/ka zna i rozumie zasady rozwiązywania problemów z wykorzystaniem zaawansowanych algorytmów i metod informatycznych
INF_K4_W05	Absolwent/ka zna i rozumie budowę oraz cykl życia przykładowych systemów informatycznych wykorzystywanych w praktyce oraz zna ograniczenia złożonych systemów informatycznych