



IV. Opis programu studiów

3. KARTA PRZEDMIOTU

Kod przedmiotu	E-EM-05-S6
Nazwa przedmiotu	Programowanie strukturalne
Nazwa przedmiotu w języku angielskim	Structured Programming
Obowiązuje od roku akademickiego	2020/21

USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	
Poziom kształcenia	
Profil studiów	
Forma i tryb prowadzenia studiów	
Zakres	
Jednostka prowadząca przedmiot	
Koordynator przedmiotu	Dr inż. Michał Łaskawski
Zatwierdził	Dziekan Wydziału Elektrotechniki Automatyki i Informatyki Dr hab. inż. Antoni Różowicz, prof. PŚk

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	
Status przedmiotu	
Język prowadzenia zajęć	
Usytuowanie modułu w planie studiów - semestr	
Wymagania wstępne	Algebra
Egzamin (TAK/NIE)	
Liczba punktów ECTS	

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	15		30		0

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Zna i rozumie zagadnienia związane z budową i funkcjonowaniem komputerowych systemów operacyjnych.	EM1_W10
	W02	Zna i rozumie zasady programowania strukturalnego oraz ma ugruntowaną wiedzę w zakresie stosowania języków programowania wysokiego poziomu.	EM1_W10, EM1_W11
Umiejętności	U01	Potrafi obsługiwać i programować komputer klasy PC.	EM1_U04
	U02	Potrafi konstruować algorytmy z wykorzystaniem podstawowych technik algorytmicznych.	EM1_U04
Kompetencje społeczne	K01	Jest gotów do krytycznej oceny posiadanej wiedzy. Rozumie potrzebę i zna możliwości ciągłego dokształcania się.	EM1_K01

TRZĘŚCI PROGRAMOWE

Forma zajęć*	Treści programowe
wykład	1. Budowa i zasady funkcjonowania komputerowych systemów operacyjnych. Historia rozwoju i podział języków programowania.
	2. Kompilator języka C i podstawowe typy danych.
	3. Wprowadzenie do preprocesora.
	4. Wyrażenia preprocesora.
	5. Wyrażenia i operatory w języku C.
	6. Instrukcje złożone warunkowe i iteracyjne w języku C.
	7. Bloki kodu, widoczność zmiennych, funkcje w języku C.
	8. Struktury i unie w języku C, wprowadzenie do wskaźników w języku C.
	9. Arytmetyka wskaźników, wskaźniki jako argumenty funkcji i wskaźniki do struktur.
	10. Zarządzanie pamięcią programu tworzonego w języku C.
	11. Obsługa plików w języku C.
	12. Zmienne dynamiczne.
	13. Implementacja wybranych algorytmów z wykorzystaniem narzędzi języka C.
	14. Wykorzystanie zestawu narzędzi, makr i skryptów powłoki do sterowania procesem kompilacji programu i analizy błędów działania programu.
	15. Wykorzystanie systemu kontroli wersji do zarządzania procesem zmian kodu źródłowego tworzonego programu.
laboratorium	1. Obsługa wybranego środowiska programistycznego języka C.
	2. Konstrukcja prostych programów w języku C i zastosowanie preprocesora.
	3. Konstrukcja złożonych programów w języku C, wykorzystujących instrukcje złożone.
	4. Wykorzystanie bloków kodu, widoczności zmiennych i funkcji.
	5. Tworzenie programów z wykorzystaniem struktur i unii języka C.
	6. Arytmetyki wskaźników.
	7. Wykorzystanie wskaźników do struktur i funkcji oraz jako argumentów funkcji.
	8. Zarządzanie pamięcią w ramach programu tworzonego w języku C.
	9. Obsługa plików z wykorzystaniem deskryptorów i strumieni do plików.
	10. Implementacja list jedno i dwukierunkowych.
	11. Implementacja kolejek, stosów i drzew.
	12. Wykorzystanie zaimplementowanych zmiennych dynamicznych.
	13. Implementacja wybranych algorytmów z wykorzystaniem narzędzi języka C.
	14. Wykorzystanie zestawu narzędzi, makr i skryptów powłoki do sterowania procesem kompilacji programu i analizy błędów działania programu.
	15. Wykorzystanie systemu kontroli wersji.

*) zostawić tylko realizowane formy zajęć

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów uczenia się					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01		X				
W02		X				
U01						X
U02						X
K01						X

FORMA I WARUNKI ZALICZENIA

Forma zajęć*	Forma zaliczenia	Warunki zaliczenia
wykład		Uzyskanie co najmniej 50% punktów z egzaminu.
laboratorium		Uzyskanie co najmniej 50% punktów z zadań realizowanych w trakcie zajęć

*) zostawić tylko realizowane formy zajęć

NAKŁAD PRACY STUDENTA

Bilans punktów ECTS							
Lp.	Rodzaj aktywności	Obciążenie studenta					Jednostka
1.	Udział w zajęciach zgodnie z planem studiów	W	C	L	P	S	h
		30		30			
3.	Inne (konsultacje, egzamin)*	4		2			h
4.	Razem przy bezpośrednim udziale nauczyciela akademickiego	66					h
5.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	2,64					ECTS
6.	Liczba godzin samodzielnej pracy studenta	34					h
7.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	1,36					ECTS
8.	Nakład pracy związany z zajęciami o charakterze praktycznym	30					h
9.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	2,00					ECTS
10.	Sumaryczne godzinowe obciążenie pracą studenta	100					h
11.	Punkty ECTS za moduł 1 punkt ECTS=25 godzin obciążenia studenta	4					

* wszelkie formy weryfikacji efektów, w tym egzaminy oraz nie więcej niż 2 godziny konsultacji dla każdej formy zajęć

LITERATURA

1. Kernighan B.W., Ritchie D.M.: Język C, WNT, Warszawa, 1988.
2. Prata S.: Język C Szkoła programowania, Helion, 2006.
3. Reese R.: Wskaźniki w języku C – przewodnik, Helion, 2014.
4. Malina W., Mironowicz P.: Programowanie strukturalne, PWN, 2018

Uwaga: wykaz literatury winien uwzględniać aktualne i dostępne publikacje