



Kierunek studiów	Informatyka
Profil	Praktyczny
Stopień studiów	2-go stopnia
Forma studiów	niestacjonarne

Sylabus przedmiotu Systemy czasu rzeczywistego

1. Dane podstawowe

Status programowy przedmiotu	Blok A: Przedmioty unikatowe dla Inżynierii Programowo-Sprzętowej
Rodzaj przedmiotu	Obligatoryjny
Kod przedmiotu	MZ-SCR-ZP
Rok studiów	2
Semestr	3
Osoba odpowiedzialna za przedmiot	dr hab. inż. Bartłomiej Kubica
Język wykładowy	polski

2. Wymiar godzin i forma zajęć

Rodzaj	Liczba godzin
Wykład	20
Laboratorium	20
Razem godzin	40

3. Cele przedmiotu

Kod	Cel
CP1	Zrozumienie istoty działania systemów działających terminowo.
CP2	Umiejętność tworzenia i analizy systemów czasu rzeczywistego.

4. Wymagania wstępne w zakresie wiedzy, umiejętności i innych kompetencji

Wymagana jest umiejętność programowania w jakimś popularnym języku - najlepiej C/C++, ew. C#, Java lub Python. Wskazana jest też podstawowa wiedza na temat systemów operacyjnych.

5. Efekty uczenia się

Wiedza

Kod	Student zna i rozumie:	Realizuje cel	Efekty kierunkowe
EU-W1	Student rozumie istotę działania systemu czasu rzeczywistego	CP1	IK7_W07
EU-W2	Student rozumie różnicę między działaniem systemu desktopowego lub serwerowego a systemem spełniającym twarde wymagania czasu rzeczywistego (hard real-time system)	CP2	IK7_W05, IK7_W07

Umiejętności

Kod	Student potrafi:	Realizuje cel	Efekty kierunkowe
EU-U1	Student umie zaprojektować system, spełniający wymagania terminowości w swoim działaniu.	CP1, CP2	IK7_U09, IK7_U20
EU-U2	Student umie wyspecyfikować wymagania systemu czasu rzeczywistego	CP2	IK7_U15, IK7_U18
EU-U3	Student umie pracować w zespole, projektującym system czasu rzeczywistego	CP1, CP2	IK7_U06

Kompetencje

Kod	Student jest gotów do:	Realizuje cel	Efekty kierunkowe
EU-K1	Student umie współpracować z innymi interesariuszami w ramach zespołu, projektującego system czasu rzeczywistego	CP1, CP2	IK7_K03, IK7_K04

6. Treści programowe

Kod	Tematyka	wykład	laboratorium	Realizuje efekt
TP1	Wprowadzenie do systemów czasu rzeczywistego (SCR). Przykłady systemów czasu rzeczywistego. Specyfika systemów czasu rzeczywistego. Definicja i model systemu czasu rzeczywistego. Reprezentacja czasu w języku C++11, w systemach POSIX i innych środowiskach. Obsługa uzależnień czasowych.	4	0	EU-K1, EU-U1, EU-U2, EU-W1, EU-W2
TP2	Szeregowanie zadań przy wymaganiach czasu rzeczywistego. Przypomnienie wybranych zagadnień programowania współbieżnego i wielowątkowego. Problemy synchronizacji. Zmienne warunkowe.	4	0	EU-U1, EU-U2, EU-W1, EU-W2
TP3	Problemy współpracy zadań przy wymaganiach czasu rzeczywistego. Zagadnienie inwersji priorytetów. Przykłady. Omówienie sytuacji Sojournera. Przegląd systemów operacyjnych czasu rzeczywistego: QNX Neutrino, VRTX, Vx Works, LynxOS, RT Linux.	4	0	EU-U1, EU-U2, EU-W1, EU-W2
TP4	Gry jako systemy czasu rzeczywistego. Różne silniki tworzenia gier. Pygame dla języka Python, biblioteka SFML dla C++, Unity dla C#. Prezentacja i porównanie wybranych silników do programowania gier	4	0	EU-U1, EU-U2, EU-W1, EU-W2
TP5	Projektowanie systemów czasu rzeczywistego. Etapy tworzenia SCR, model cyklu opracowania SCR, cykl życia oprogramowania. Modele tworzenia oprogramowania SCR. Sieci Petriego jako narzędzie modelowania systemów czasu rzeczywistego.	4	0	EU-K1, EU-U1, EU-U3, EU-W2
TP6	Analiza wymagań systemu czasu rzeczywistego. Omówienie i konsultacje studentów w sprawie ich projektów.	0	4	EU-K1, EU-U2, EU-U3, EU-W2
TP7	Projekt wstępny zadanego systemu. Konsultacje studentów i ocena wstępnego projektu.	0	4	EU-K1, EU-U1, EU-U3
TP8	Projekt szczegółowy systemu czasu rzeczywistego. Konsultacje studentów, ich prezentacje i modyfikacja projektów po dyskusji.	0	4	EU-K1, EU-U1, EU-U2, EU-U3, EU-W1
TP9	Implementacja systemu czasu rzeczywistego. Prezentacje studentów i ocena kolejnej części projektu.	0	4	EU-K1, EU-U3, EU-W1, EU-W2
TP10	Integracja i testowanie zaimplementowanego systemu czasu rzeczywistego. Końcowa ocena projektów studentów.	0	4	EU-W1, EU-W2

Razem godzin: 40

7. Metody kształcenia

Kod	Metoda
MK1	Wykład zdalny za pomocą MS Teams.
MK2	dyskusja
MK3	wykład wsparty prezentacją komputerową
MK4	analiza przypadków
MK5	miniprojekt zespołowy realizowany na zajęciach

8. Nakład pracy studenta

Aktywność studenta	Obciążenie
Praca nad projektem - indywidualna i zespołowa	40
Przygotowanie do egzaminu	20
Praca związana z: laboratorium	20
Praca związana z: wykład	20
Liczba punktów ECTS (1 punkt=25h)	4
Procentowy udział pracy własnej studenta w sumarycznym obciążeniu studenta	60,00%
Sumaryczne obciążenie pracą studenta	100

9. Status zaliczenia przedmiotu

Egzamin za pomocą systemu Inspira lub Moodle.

Forma studiów	Egzamin	Praca egzaminacyjna	Zaliczenie	Praca zaliczeniowa
niestacjonarne	×			

10. Metody weryfikacji efektów uczenia się

Składowe oceny końcowej

Forma sprawdzenia	Wybrana forma	Punktacja	Realizuje efekt
Egzamin pisemny	×	60	EU-U2, EU-U1, EU-W2, EU-W1
Egzamin ustny			
Sprawdzian pisemny			
Zaliczeniowy przegląd prac			
Referat pisemny			
Referat ustny			
Kolokwium			
Praca domowa			
Miniprojekt			
Praca na zajęciach	×	10	EU-K1, EU-U3, EU-U2, EU-U1
Projekt z dokumentacją	×	30	EU-K1, EU-U3, EU-U2, EU-U1
Ustna prezentacja projektu			
Obecność na zajęciach			
Sprawdzian ustny			
Kartkówka			
Aktywność na zajęciach			
Egzaminacyjny przegląd prac			
Sprawozdanie z praktyki zawodowej			
Prezentacja indywidualna			
Prezentacja zespołowa			

Zasady wyliczania oceny z przedmiotu

Zakres punktów	Ocena
0 – 40	2,0
41 – 50	3,0
51 – 60	3,5
61 – 70	4,0
71 – 80	4,5
81 – 100	5,0

11. Macierz realizacji przedmiotu

Efekt uczenia się	Cel przedmiotu	Treści programowe	Metody kształcenia
EU-W1	CP1	TP1, TP2, TP3, TP4, TP8, TP9, TP10	MK1, MK2, MK3, MK4, MK5
EU-W2	CP2	TP1, TP2, TP3, TP4, TP5, TP6, TP9, TP10	MK1, MK2, MK3, MK4, MK5

Efekt uczenia się	Cel przedmiotu	Treści programowe	Metody kształcenia
EU-U1	CP1, CP2	TP1, TP2, TP3, TP4, TP5, TP7, TP8	MK1, MK2, MK3, MK4, MK5
EU-U2	CP2	TP1, TP2, TP3, TP4, TP6, TP8	MK1, MK2, MK3, MK4, MK5
EU-U3	CP1, CP2	TP5, TP6, TP7, TP8, TP9	MK1, MK2, MK3, MK4, MK5
EU-K1	CP1, CP2	TP1, TP5, TP6, TP7, TP8, TP9	MK1, MK2, MK3, MK4, MK5

12. Odniesienie efektów uczenia się

Efekt uczenia się	Efekty kształcenia dla kierunku studiów	Charakterystyki drugiego stopnia w obszarze kształcenia
EU-W1	IK7_W07	P7S_WG
EU-W2	IK7_W05, IK7_W07	P7S_WG
EU-U1	IK7_U09, IK7_U20	P7S_UW
EU-U2	IK7_U15, IK7_U18	P7S_UW
EU-U3	IK7_U06	P7S_UK
EU-K1	IK7_K03, IK7_K04	P7S_KO, P7S_KR

13. Literatura

Literatura podstawowa

1. Robert Love, Linux. Programowanie systemowe., Helion

Literatura uzupełniająca

1. Paweł Majdzik, Programowanie współbieżne. Systemy czasu rzeczywistego., Helion

Pozostałe

1. Bartłomiej Kubica, Slajdy z wykładu

14. Informacje o nauczycielach akademickich

Osoby odpowiedzialne za przedmiot

1. dr hab. inż. Bartłomiej Kubica

Osoby prowadzące przedmiot

1. dr hab. inż. Bartłomiej Kubica