

I. KARTA PRZEDMIOTU

1. Nazwa przedmiotu: **SYSTEMY CZASU RZECZYWISTEGO**
2. Kod przedmiotu: **Scr**
3. Jednostka prowadząca: **Wydział Mechaniczno-Elektryczny**
4. Kierunek: **Mechatronika**
5. Specjalność: **Techniki Komputerowe w Mechatronice**
6. Moduł: **treści specjalnościowych**
7. Poziom studiów: **I stopnia**
8. Forma studiów: **stacjonarne**
9. Semestr studiów: **VII**
10. Profil: **praktyczny**
11. Prowadzący: **mgr inż. Marcin Wiśniewski**

CEL PRZEDMIOTU

C1	Zapoznanie studentów z podstawami systemów czasu rzeczywistego.
C2	Zapoznanie studentów z architekturą wielowątkową i wieloprocesową
C3	Zapoznanie studentów z narzędziami wykorzystywanymi przy implementacji aplikacji sterowanych czasem

WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

1	Umiejętność programowanie w języku C oraz C++
----------	---

EFEKTY KSZTAŁCENIA

EK1	Student zna cechy charakterystyczne systemu czasu rzeczywistego oraz potrafi dokonać ich kategoryzacji
EK2	Student zna pojęcia procesu oraz wątku, mechanizmy synchronizacji oraz komunikacji stosowane w RTLinux
EK3	Student zna funkcje RTLinux pozwalające na odczyt czasu systemowego, wprowadzanie opóźnień oraz wykorzystanie timer-ów
EK4	Student potrafi zaimplementować proces wielowątkowy i dokonać synchronizacji równoległe wykonywanych wątków
EK5	Student potrafi wykorzystać mechanizmy komunikacji i synchronizacji międzyprocesowej wykorzystywane w RTLinux
EK6	Student potrafi zaimplementować aplikację sterowaną czasem
EK7	Student potrafi stworzyć i skompilować program w RTLinux

TREŚCI PROGRAMOWE

WYKŁADY		Liczba godzin
W1	Wprowadzenie do systemów czasu rzeczywistego.	2
W2	Ogólna prezentacja systemu RTLinux	2
W3	Środowisko bash make i gcc	2
W4	Wątki i procesy w RTLinux. Mechanizmy komunikacji i synchronizacji wątków/procesów w RTLinux.	2
W5	Timery i zdarzenia.	2
Razem		10
ĆWICZENIA		
Ć1	Zaliczenie	4
Razem		4
ZAJĘCIA LABORATORYJNE		
L1	Zapoznanie ze środowiskiem RTLinux	2

L3	Aplikacje wieloprotokółowe, synchronizacja i komunikacja międzyprocesowa.	6
L4	Zastosowanie timerów.	4

Razem **16**

NARZĘDZIA DYDAKTYCZNE

1	Notebook z projektorem
2	Tablica i kolorowe pisaki
3	Stanowiska komputerowe z oprogramowaniem dydaktycznym
4	Płytki Raspberry PI

SPOSOBY OCENY

FORMUJĄCA

F1	Odpowiedź ustna	EK1-EK7
F2	Implementacja + sprawozdanie	EK4-EK7

PODSUMOWUJĄCA

P1	Zaliczenie - zadanie do pracy w zespołach 2 osobowych	EK1-EK7
-----------	---	---------

OBCIĄŻENIE PRACĄ STUDENTA

Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności	
	semestr	razem
udział w wykładach	10	10
udział w ćwiczeniach	4	4
udział w zajęciach laboratoryjnych	16	16
SUMA GODZIN W SEMESTRZE	30	30
PUNKTY ECTS W SEMESTRZE	2	2

LITERATURA

PODSTAWOWA

1	https://en.wikipedia.org/wiki/RTLinux
----------	---

PROWADZĄCY PRZEDMIOT

1	mgr inż. Marcin Wiśniewski, marcin.wisniewski@ctm.gdynia.pl
----------	---

Formy oceny

Efekt	Na ocenę 2	Na ocenę 3	Na ocenę 4	Na ocenę 5
EK1	<i>Student zna cechy charakterystyczne systemu czasu rzeczywistego oraz potrafi dokonać ich kategoryzacji</i>			
	Student nie potrafi podać cech charakterystycznych systemu czasu rzeczywistego	Student potrafi podać cechy charakterystyczne systemu czasu rzeczywistego. Nie potrafi podać przykładów takich systemów.	Student potrafi podać cechy charakterystyczne systemu czasu rzeczywistego. Potrafi podać przykłady takich systemów. Umie dokonać ich kategoryzacji ze względu na wybraną cechę.	Student potrafi podać cechy charakterystyczne systemu czasu rzeczywistego. Potrafi podać przykłady takich systemów. Umie dokonać ich pełnej kategoryzacji.
EK2	<i>Student zna pojęcia procesu oraz wątku, mechanizmy synchronizacji oraz komunikacji stosowane w RTLinux</i>			
	Student:1. Nie potrafi odróżnić procesu od wątku. 2. Nie wie jak tworzy się wątki/procesy w RTLinux. 3. Nie zna mechanizmów synchronizacji wątków/procesów w RTLinux.4. Nie zna mechanizmów komunikacji wątków/procesów w RTLinux.	Student:1. Potrafi odróżnić procesu od wątku. 2. Wie jak tworzy się wątki/procesy w RTLinux. 3. Nie zna mechanizmów synchronizacji wątków/procesów w RTLinux.4. Nie zna mechanizmów komunikacji wątków/procesów w RTLinux.	Student:1. Potrafi odróżnić procesu od wątku. 2. Wie jak tworzy się wątki/procesy w RTLinux. 3. Zna mechanizmy synchronizacji wątków/procesów w RTLinux.4. Nie zna mechanizmów komunikacji wątków/procesów w RTLinux.	Student:1. Potrafi odróżnić procesu od wątku. 2. Wie jak tworzy się wątki/procesy w RTLinux. 3. Zna mechanizmy synchronizacji wątków/procesów w RTLinux.4. Zna mechanizmy komunikacji wątków/procesów w RTLinux.
EK3	<i>Student zna funkcje RTLinux pozwalające na odczyt czasu systemowego, wprowadzanie opóźnień oraz wykorzystanie timer-ów</i>			
	Student (wszystkie wiadomości odnoszą się do systemu RTLinux)1. Nie zna funkcji odczytu czasu systemowego. 2. Nie zna funkcji sterowania czasem systemowym.3. Nie zna metody tworzenia timerów.4. Nie zna sposobu wykorzystania timerów do sterowania przebiegiem aplikacji.	Student (wszystkie wiadomości odnoszą się do systemu RTLinux)1. Zna funkcję odczytu czasu systemowego. 2. Zna funkcję sterowania czasem systemowym.3. Nie zna metody tworzenia timerów.4. Nie zna sposobu wykorzystania timerów do sterowania przebiegiem aplikacji.	Student (wszystkie wiadomości odnoszą się do systemu RTLinux)1. Zna funkcję odczytu czasu systemowego. 2. Zna funkcję sterowania czasem systemowym.3. Zna metody tworzenia timerów.4. Nie zna sposobu wykorzystania timerów do sterowania przebiegiem aplikacji.	Student (wszystkie wiadomości odnoszą się do systemu RTLinux)1. Zna funkcję odczytu czasu systemowego. 2. Zna funkcję sterowania czasem systemowym.3. Zna metody tworzenia timerów.4. Zna sposoby wykorzystania timerów do sterowania przebiegiem aplikacji.
EK4	<i>Student potrafi zaimplementować proces wielowątkowy i dokonać synchronizacji równoległe wykonywanych wątków</i>			
	Student 1. Nie potrafi tworzyć nowych wątków. 2. Nie potrafi określać stanu wątków i definiować ich parametrów. 3. Nie potrafi zaimplementować wymiany informacji pomiędzy wątkami. 4. Nie potrafi synchronizować równoległe wykonywanych wątków.	Student 1. Potrafi tworzyć nowe wątki. 2. Potrafi określać stan wątków i definiować ich parametry. 3. Nie potrafi zaimplementować wymiany informacji pomiędzy wątkami. 4. Nie potrafi synchronizować równoległe wykonywanych wątków.	Student 1. Potrafi tworzyć nowe wątki. 2. Potrafi określać stan wątków i definiować ich parametry. 3. Potrafi zaimplementować wymianę informacji pomiędzy wątkami. 4. Nie potrafi synchronizować równoległe wykonywanych wątków.	Student 1. Potrafi tworzyć nowe wątki. 2. Potrafi określać stan wątków i definiować ich parametry. 3. Potrafi zaimplementować wymianę informacji pomiędzy wątkami. 4. Potrafi synchronizować równoległe wykonywane wątki.

EK5	<i>Student potrafi wykorzystać mechanizmy komunikacji i synchronizacji międzyprocesowej wykorzystywane w RTLinux</i>			
	Student 1. Nie potrafi rejestrować procesów w systemie oraz wyszukiwać uruchomionych procesów i nawiązywać z nimi połączenie 2. Nie potrafi stosować mechanizmów komunikacji międzyprocesowej 3. Nie potrafi stosować mechanizmów synchronizacji międzyprocesowej	Student 1. Potrafi rejestrować procesy w systemie oraz wyszukiwać uruchomione procesy i nawiązywać z nimi połączenie 2. Nie potrafi stosować mechanizmów komunikacji międzyprocesowej 3. Nie potrafi stosować mechanizmów synchronizacji międzyprocesowej	Student 1. Potrafi rejestrować procesy w systemie oraz wyszukiwać uruchomione procesy i nawiązywać z nimi połączenie 2. Potrafi stosować mechanizmy komunikacji międzyprocesowej 3. Nie potrafi stosować mechanizmów synchronizacji międzyprocesowej	Student 1. Potrafi rejestrować procesy w systemie oraz wyszukiwać uruchomione procesy i nawiązywać z nimi połączenie 2. Potrafi stosować mechanizmy komunikacji międzyprocesowej 3. Potrafi stosować mechanizmy synchronizacji międzyprocesowej
EK6	<i>Student potrafi zaimplementować aplikację sterowaną czasem</i>			
	Student 1. Nie potrafi używać funkcji "czasowych" (np. odczyt, konwersja, interwał) 2. Nie potrafi tworzyć, usuwać oraz modyfikować timer-ów 3. Nie potrafi używać timer-ów do sterowania aplikacją	Student 1. Potrafi używać funkcji "czasowych" (np. odczyt, konwersja, interwał) 2. Nie potrafi tworzyć, usuwać oraz modyfikować timer-ów 3. Nie potrafi używać timer-ów do sterowania aplikacją	Student 1. Potrafi używać funkcji "czasowych" (np. odczyt, konwersja, interwał) 2. Potrafi tworzyć, usuwać oraz modyfikować timer-y 3. Nie potrafi używać timer-ów do sterowania aplikacją	Student 1. Potrafi używać funkcji "czasowych" (np. odczyt, konwersja, interwał) 2. Potrafi tworzyć, usuwać oraz modyfikować timer-y 3. Potrafi używać timer-ów do sterowania aplikacją
EK7	<i>Student potrafi stworzyć i skompilować program w RTLinux</i>			
	Student1. Nie potrafi tworzyć projektów, określać ich parametrów2. Nie potrafi dodawać, usuwać, modyfikować: klasy, pliki, katalogi3. Nie potrafi posługiwać się edytorem tekstu, wyszukiwać tekstu, ustalać parametrów edytora4. Nie potrafi kompilować aplikacji	Student1. Potrafi tworzyć projekty, określać ich parametry2. Potrafi dodawać, usuwać, modyfikować: klasy, pliki, katalogi3. Nie potrafi posługiwać się edytorem tekstu, wyszukiwać tekstu, ustalać parametrów edytora4. Nie potrafi kompilować aplikacji oraz używać jej w maszynie wirtualnej	Student1. Potrafi tworzyć projekty, określać ich parametry2. Potrafi dodawać, usuwać, modyfikować: klasy, pliki, katalogi3. Potrafi posługiwać się edytorem tekstu, wyszukiwać tekst, ustalać parametry edytora4. Nie potrafi kompilować aplikacji oraz używać jej w maszynie wirtualnej	Student1. Potrafi tworzyć projekty, określać ich parametry2. Potrafi dodawać, usuwać, modyfikować: klasy, pliki, katalogi3. Potrafi posługiwać się edytorem tekstu, wyszukiwać tekst, ustalać parametry edytora4. Potrafi kompilować aplikację oraz używać jej