

# KARTA PRZEDMIOTU

## 1. Informacje ogólne

<b>Nazwa przedmiotu i kod (wg planu studiów):</b>	Programowanie aplikacji współbieżnych w systemie QNX <b>D1.13</b>
<b>Nazwa przedmiotu (j. ang.):</b>	Designing concurrent applications in QNX
<b>Kierunek studiów:</b>	Informatyka
<b>Specjalność/specjalizacja:</b>	Informatyka praktyczna
<b>Poziom kształcenia:</b>	studia I stopnia
<b>Profil kształcenia:</b>	praktyczny (P)
<b>Forma studiów:</b>	studia stacjonarne / studia niestacjonarne
<b>Obszar kształcenia:</b>	nauki techniczne
<b>Dziedzina:</b>	nauki techniczne
<b>Dyscyplina nauki:</b>	Informatyka
<b>Koordinator przedmiotu:</b>	mgr Radosław Gołąb

## 2. Ogólna charakterystyka przedmiotu

<b>Przynależność do modułu:</b>	kształcenie specjalnościowe
<b>Status przedmiotu:</b>	obieralny
<b>Język wykładowy:</b>	Polski
<b>Rok studiów, semestr:</b>	III, 5
<b>Forma i wymiar zajęć według planu studiów:</b>	stacjonarne - wykład 15 h, ćw. laboratoryjne 30 h niestacjonarne - wykład 15 h, ćw. laboratoryjne 15 h
<b>Interesariusze i instytucje partnerskie (nieobowiązkowe)</b>	
<b>Wymagania wstępne / Przedmioty wprowadzające:</b>	Język C++ / Programowanie I

### 3. Bilans punktów ECTS

Całkowita liczba punktów ECTS: (A + B)		5	stacjonarne	Niestacjonarne
<b>A. Liczba godzin wymagających bezpośredniego udziału nauczyciela z podziałem na typy zajęć oraz całkowita liczba punktów ECTS osiągniętych na tych zajęciach:</b>	obecność na wykładach		30	15
	obecność na ćwiczeniach laboratoryjnych		30	15
	udział w konsultacjach		7	7
	<b>w sumie:</b> ECTS		67 2,6	37 1,4
<b>B. Poszczególne typy zadań do samokształcenia studenta (nie-wymagających bezpośredniego udziału nauczyciela) wraz z planowaną średnią liczbą godzin na każde i sumaryczną liczbą ECTS:</b>	przygotowanie do ćwiczeń laboratoryjnych		10	10
	wykonanie sprawozdań		20	25
	przygotowanie do kolokwium		10	15
	praca w sieci		5	10
	przygotowanie do konsultacji		5	10
	uzupełnienie/studiowanie notatek		5	10
	studiowanie zalecanej literatury		5	10
	<b>w sumie:</b> ECTS		60 2,4	90 3,6
<b>C. Liczba godzin praktycznych / laboratoryjnych w ramach przedmiotu oraz związana z tym liczba punktów ECTS:</b>	udział w ćwiczeniach laboratoryjnych		30	15
	praca praktyczna samodzielna		30	45
	<b>w sumie:</b> ECTS		60 2	60 2

### 4. Opis przedmiotu

<b>Cel przedmiotu:</b> Celem zajęć jest poznanie budowy, oraz zasad funkcjonowania oprogramowania spełniającego wymagania czasu rzeczywistego, na przykładzie aplikacji dla systemu czasu rzeczywistego QNX.
<b>Metody dydaktyczne:</b> <i>wykład, praktyczne ćwiczenia laboratoryjne</i>
<b>Treści kształcenia:</b> <b>Wykłady:</b> <ol style="list-style-type: none"> <li>1. System czasu rzeczywistego, klasyfikacja zadań systemu, ograniczenia czasowe, ograniczenia ostre (hard) i łagodne (soft), szeregowanie zadań, podstawowe koncepcje implementacyjne.</li> <li>2. Wymagania stawiane systemom operacyjnym czasu rzeczywistego, standard POSIX, praca wielozadaniowa, modele budowy systemu, szeregowanie zadań, zdarzenia i sygnały, synchronizacja i komunikacja zadań, zakleszczenia i inwersja priorytetu, pamięć wspólna, uzależnienia czasowe, dostęp do urządzeń.</li> <li>3. Przykład systemu QNX: architektura systemu, praca w konfiguracji rozproszonej, implementacja architektury klient-serwer. Omówienie metod i narzędzi do tworzenia oprogramowania czasu rzeczywistego w dziedzinie klasycznych systemów informatycznych.</li> <li>4. Specyfika systemów czasu rzeczywistego (współbieżność, wymagania terminowości odpowiedzi, ciągłość działania, przewidywalność i niezawodność)</li> <li>5. Zagadnienia analizy wymagań, specyfikacji i projektowania prostych systemów czasu rzeczywistego.</li> <li>6. Przykłady sieci Petriego jako podstawowego narzędzia opisu. Zastosowanie sieci Petriego do precyzyjnego oraz abstrakcyjnego opisu własności projektowanego systemu.</li> <li>7. Przegląd strukturalnych metod tworzenia oprogramowania.</li> </ol> <b>Ćwiczenia laboratoryjne:</b>

1. Cechy specyficzne systemów czasu rzeczywistego.
2. Zasady współpracy zadań i problemy z niej wynikające.
3. Poznanie budowy, oraz zasad budowy oprogramowania spełniającego wymagania czasu rzeczywistego.
4. Podstawowy moduł czasu rzeczywistego.
5. Komunikacja z programami w trybie użytkownika z wykorzystaniem kolejek FIFO i pamięci współdzielonej.
6. Aplikacje oparte na wątkach a system czasu rzeczywistego.
7. Diagnostyka urządzenia fizycznego z wykorzystaniem systemu czasu rzeczywistego.
8. Aplikacje oparte na rozwiązaniach komunikacji sieciowej.

## 5. Efekty kształcenia i sposoby weryfikacji

Efekty kształcenia				
Efekt przedmiotu (kod przedmiotu + kod efektu kształcenia)		Student, który zaliczył przedmiot (spełnił minimum wymagań)	Efekt kierunkowy	
D1.13_W01		<b>Wiedza:</b> 1. Zna charakterystykę i podstawowe struktury systemów czasu rzeczywistego. 2. Zna zasadę działania systemu operacyjnego. 3. Zna wzorce oprogramowania systemów czasu rzeczywistego	K_W07	
D1.13_W02			K_W08	
D1.13_W03			K_W18	
D1.13_U01		<b>Umiejętności</b> 1. Potrafi dokonać konfiguracji swojego środowiska pracy, określać wykorzystanie przez niego zasobów oraz wykorzystywać zasoby systemu dla potrzeb własnych procesów. 2. Zna wzorce oprogramowania systemów czasu rzeczywistego 3. Wie z jakich elementów składa się system czasu rzeczywistego	K_U12	
D1.13_U02			K_U17	
D1.13_U03			K_U25	
D1.13_K01		<b>Kompetencje społeczne</b> 1. Ma świadomość roli i znaczenia systemów czasu rzeczywistego w przedsiębiorstwie, gospodarce i społeczeństwie 2. Student rozumie potrzebę wykorzystania nabytej wiedzy na niezwykle szybko rozwijającym się rynku aplikacji.	K_K03	
D1.13_K02			K_K08	
<b>Sposoby weryfikacji efektów kształcenia:</b> <i>(np. dyskusja, gra dydaktyczna, zadanie e-learningowe, ćwiczenie laboratoryjne, projekt indywidualny/ grupowy, zajęcia terenowe, referat studenta, praca pisemna, kolokwium, test zaliczeniowy, egzamin, opinia eksperta zewnętrznego, etc. Dodać do każdego wybranego sposobu symbol zakładanego efektu, jeśli jest ich więcej)</i>				
Lp.	Efekt przedmiotu	Sposób weryfikacji	Ocena formująca	Ocena końcowa
1	D1.13_W01 D1.13_W02 D1.13_W03 D1.13_U01 D1.13_U02 D1.13_U03	kolokwium zaliczeniowe	ocena z kolokwium - sprawdzian wiedzy i umiejętności	Ocena końcowa z laboratorium - średnia z ocen formujących
2	D1.13_U01 D1.13_U02 D1.13_U03 D1.13_K01 D1.13_K02	ćwiczenia laboratoryjne	ocena sprawozdania z prac laboratoryjnych, ocena zaangażowania na zajęciach	
<b>Kryteria oceny</b> (oceny 3,0 powinny być równoważne z efektami kształcenia, choć mogą być bardziej szczegółowo opisane):				

w zakresie wiedzy		Efekt kształcenia
Na ocenę 3,0	<p>Student uzyskał min. 50% wymaganej wiedzy w zakresie obowiązującego materiału. Student:</p> <ul style="list-style-type: none"> <li>- Zna zasadę działania systemu operacyjnego.</li> <li>- Zna wzorce oprogramowania systemów czasu rzeczywistego</li> <li>- Zna modele analizy i projektowania systemów czasu rzeczywistego.</li> </ul>	D1.13_W01 D1.13_W02 D1.13_W03
Na ocenę 5,0	<p>Student zdobył powyżej 95% wymaganej wiedzy w zakresie obowiązującego materiału. Student:</p> <ul style="list-style-type: none"> <li>- Potrafi dokonać konfiguracji swojego środowiska pracy, określać wykorzystanie przez niego zasobów oraz wykorzystywać zasoby systemu dla potrzeb własnych procesów.</li> <li>- Wie jak tworzyć aplikacje czasu rzeczywistego w oparciu o wzorce.</li> <li>- Wie z jakich elementów składa się system czasu rzeczywistego</li> </ul>	D1.13_W01  D1.13_W02 D1.13_W03
w zakresie umiejętności		Efekt kształcenia
Na ocenę 3,0	<p>Student uzyskał min. 50% wymaganych umiejętności w zakresie obowiązującego materiału. Student potrafi:</p> <ul style="list-style-type: none"> <li>- Potrafi budować proste modele systemów czasu rzeczywistego.</li> <li>- Umie wyjaśnić aspekty związane z wdrożeniem systemu czasu rzeczywistego.</li> <li>- Ma podstawowe umiejętności w zakresie tworzenia architektury, funkcjonowania i programowania aplikacji w systemach czasu rzeczywistego.</li> </ul>	D1.13_U01 D1.13_U02  D1.13_U03
Na ocenę 5,0	<p>Student uzyskał min. 50% wymaganych umiejętności w zakresie obowiązującego materiału. Student potrafi:</p> <ul style="list-style-type: none"> <li>- Student potrafi napisać i przeanalizować aplikację czasu rzeczywistego.</li> <li>- Student potrafi samodzielnie zaimplementować procesy czasu rzeczywistego.</li> <li>- Umie zapewnić komunikację pomiędzy procesami w systemie czasu rzeczywistego</li> </ul>	D1.13_U01 D1.13_U02  D1.13_U03
w zakresie kompetencji społecznych		Efekt kształcenia
Na ocenę 3,0	Student osiągnął wymagane kompetencje społeczne na poziomie min. 50%.	D1.13_K01 D1.13_K02
Na ocenę 5,0	Student osiągnął wymagane kompetencje społeczne na poziomie wyższym niż 90%.	D1.13_K01 D1.13_K02

**Student, który nie osiągnął zakładanych efektów kształcenia, nie zalicza przedmiotu.**

**Kryteria oceny końcowej:**

ocena z laboratorium:  
ocena z kolokwium: 30 %  
ocena ze sprawozdania: 50%  
samodzielne wykonanie ćwiczeń laboratoryjnych: 15%  
aktywność na zajęciach: 5%

**Zalecana literatura :****Literatura podstawowa:**

1. Jaskiewicz, Andrzej, Inżynieria oprogramowania, Gliwice, Helion, 1997.
2. Lal K., Rak T., Orkisz K., RTLinux - system czasu rzeczywistego. Gliwice, Helion, 2003.
3. Dariusz Bismor, Programowanie systemów sterowania: narzędzia i metody, Warszawa, Wydawnictwa Naukowo-Techniczne, 2010.

**Literatura uzupełniająca:**

1. Keith Haviland, Dina Gray, Ben Salama, Unix - programowanie systemowe, Warszawa , "RM", 1999

**Informacje dodatkowe:****Dodatkowe obowiązki prowadzącego wraz z szacowaną całkowitą liczbą godzin:**

Konsultacje – 15 godzin

Poprawa prac projektowych – 10 godzin

Przygotowanie ćwiczeń laboratoryjnych - 5 godzin

W sumie: 30 godzin