



### KARTA MODUŁU / KARTA PRZEDMIOTU

Kod modułu	
Nazwa modułu	<b>Programowanie systemów rozproszonych</b>
Nazwa modułu w języku angielskim	<b>Programming of distributed systems</b>
Obowiązuje od roku akademickiego	<b>2012/2013</b>

### A. USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	<b>Informatyka</b>
Poziom kształcenia	<b>II stopień</b> (I stopień / II stopień)
Profil studiów	<b>Ogólnoakademicki</b> (ogólnoakademicki / praktyczny)
Forma i tryb prowadzenia studiów	<b>niestacjonarne</b> (stacjonarne / niestacjonarne)
Specjalność	<b>Systemy informacyjne</b>
Jednostka prowadząca moduł	<b>Katedra Informatyki</b>
Koordinator modułu	<b>Robert Tomaszewski</b>
Zatwierdził:	

### B. OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	<b>Kierunkowy</b> (podstawowy / kierunkowy / inny HES)
Status modułu	<b>Obowiązkowy</b> (obowiązkowy / nieobowiązkowy)
Język prowadzenia zajęć	<b>polski</b>
Usytuowanie modułu w planie studiów - semestr	<b>1</b>
Usytuowanie realizacji przedmiotu w roku akademickim	<b>Semestr zimowy</b> (semestr zimowy / letni)
Wymagania wstępne	<b>Programowanie w języku C 2</b> <b>Architektura systemów komputerowych 1</b> (kody modułów / nazwy modułów)
Egzamin	<b>tak</b> (tak / nie)
Liczba punktów ECTS	<b>5</b>

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	inne
w semestrze	<b>18</b>		<b>9</b>	<b>9</b>	



### C. EFEKTY KSZTAŁCENIA I METODY SPRAWDZANIA EFEKTÓW KSZTAŁCENIA

<b>Cel modułu</b>	Usystematyzowanie systemów rozproszonych, poznanie podstawowych problemów konstruowania i programowania (komunikacji i synchronizacji), omówienie wybranych algorytmów rozproszonych/współbieżnych, nabycie umiejętności poprawnej synchronizacji pracy wątków i procesów (3-4 linijki)
-------------------	--

Symbol efektu	Efekty kształcenia	Forma prowadzenia zajęć (w/ć/lp/inne)	odniesienie do efektów kierunkowych	odniesienie do efektów obszarowych
W_01	Znajomość problematyki konstruowania (architektura) oraz programowania (komunikacja i synchronizacja) systemów rozproszonych/przetwarzających równoległe.	W	K_W02 K_W03 K_W09	T2A_W03 T2A_W04 T2A_W07
W_02	Wiedza na temat dowodzenia poprawności programów współbieżnych. Znajomość aparatu matematyczno-pojęciowego.	W	K_W02	T2A_W03 T2A_W07
W_03	Poznanie różnych mechanizmów synchronizowania przetwarzania rozproszonego, w szczególności API wielozadaniowego systemu operacyjnego (język C).	W, L	K_W02 K_W03	T2A_W03 T2A_W07
U_01	Umiejętność rozwiązywania typowych problemów programowania współbieżnego.	L	K_U01 K_U07	T2A_U01 T2A_U08
U_02	Umiejętność przekształcania problemu programistycznego do postaci przetwarzania równoległego i poprawnej synchronizacji poznanymi mechanizmami.	L, P	K_U01	T2A_U01
K_01	Praca w zespole.	P	K_U01 K_U02 K_U03 K_K02	T2A_U01 T2A_U02 T2A_U03 T2A_K03

### Treści kształcenia:

#### 1. Treści kształcenia w zakresie wykładu

Nr wykładu	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1	Podział systemów rozproszonych/równoległych – taksonomia Flynna. Przykładowe architektury multikomputerów i zasada ich działania.	W_01
2	Architektury wieloprocesorów: SMP, NUMA, COMA. Problem spójności pamięci podręcznych i skalowalność.	W_01
3	Programowanie wieloprocesorów – pojęcie wątku i procesu, zalety i wady.	W_01 W_03
4	Programowanie wieloprocesorów – podstawy API Microsoft Windows w zakresie wielowątkowości. Metody synchronizacji: współdzielona zmienna globalna, obiekt sekcji krytycznej, semafor, obiekt zdarzenia.	W_01 W_03
5	Algorytmy współbieżne, warunki poprawności programu współbieżnego, pojęcie przeplotu, blokady, zagłodzenia, hazardu (wyścigu) oraz wzajemnego wykluczania.	W_02
6-7	Modelowanie problemu wzajemnego wykluczania (synchronizacja). Algorytm Dekkera.	W_02
8	Problem producenta-konsumenta: analiza, modele i przykładowe rozwiązanie w języku C.	W_02 W_03 U_01
9	Problem uczujących filozofów: analiza, modele i przykładowe rozwiązanie w	W_02



	języku C.	W_03 U_01
--	-----------	--------------

### 2. Treści kształcenia w zakresie ćwiczeń

Nr zajęć ćwicz.	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1		
2		
3		
4		

### 3. Treści kształcenia w zakresie zadań laboratoryjnych

Nr zajęć lab.	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1	Implementacja programu sortującego przez scalanie z podziałem na podprogramy.	U_01 U_02
2	Zrównoleglenie programu sortującego przez scalanie (poprzez wielowątkowość) z synchronizacją za pomocą zmiennych globalnych.	U_01 U_02
3	Zrównoleglenie programu sortującego przez scalanie (poprzez wielowątkowość) z synchronizacją za pomocą obiektów sekcji krytycznej.	U_01 U_02
4	Zrównoleglenie programu sortującego przez scalanie (poprzez wielowątkowość) z synchronizacją za pomocą semaforów oraz obiektów zdarzeń.	U_01 U_02
5	Implementacja programu rozwiązującego problem 2ch producentów i 2ch konsumentów współdzielących ten sam bufor.	U_01 U_02

### 4. Charakterystyka zadań projektowych

Projekt realizowany w zespołach 2-3 osobowych zespołach. Tematyka obejmuje równoległą implementację zadanego problemu, reprezentującego konkretny problem obliczeniowy lub symulację rzeczywistej sytuacji (np. ruch drogowy), w postaci programu wielowątkowego w dowolnym języku programowania wspierającym wielowątkowość. Weryfikacja poprawności implementacji poprzez symulację (brak blokad i zagłódzeń). W sprawozdaniu wymagany opis i uzasadnienie wybranej implementacji i mechanizmu synchronizacji.

### 5. Charakterystyka zadań w ramach innych typów zajęć dydaktycznych

## Metody sprawdzania efektów kształcenia

Symbol efektu	Metody sprawdzania efektów kształcenia (sposób sprawdzenia, w tym dla umiejętności – odwołanie do konkretnych zadań projektowych, laboratoryjnych, itp.)
W_01	Egzamin
W_02	Egzamin
W_03	Egzamin, kolokwium (z zakresu zajęć laboratoryjnych)
U_01	Kolokwium
U_02	Kolokwium, ocena z projektu
K_01	Ocena z projektu



### D. NAKŁAD PRACY STUDENTA

Bilans punktów ECTS		
	Rodzaj aktywności	obciążenie studenta
1	Udział w wykładach	<b>18</b>
2	Udział w ćwiczeniach	
3	Udział w laboratoriach	<b>9</b>
4	Udział w konsultacjach (2-3 razy w semestrze)	<b>2</b>
5	Udział w zajęciach projektowych	<b>9</b>
6	Konsultacje projektowe	<b>9</b>
7	Udział w egzaminie	<b>2</b>
8		
9	<b>Liczba godzin realizowanych przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>51</b> <i>(suma)</i>
10	<b>Liczba punktów ECTS, którą student uzyskuje na zajęciach wymagających bezpośredniego udziału nauczyciela akademickiego</b> <i>(1 punkt ECTS=25-30 godzin obciążenia studenta)</i>	<b>2</b>
11	Samodzielne studiowanie tematyki wykładów	<b>9</b>
12	Samodzielne przygotowanie się do ćwiczeń	
13	Samodzielne przygotowanie się do kolokwium	
14	Samodzielne przygotowanie się do laboratoriów	<b>5</b>
15	Wykonanie sprawozdań	
15	Przygotowanie do kolokwium końcowego z laboratorium	<b>4</b>
17	Wykonanie projektu lub dokumentacji	<b>20</b>
18	Przygotowanie do egzaminu	<b>15</b>
19		
20	<b>Liczba godzin samodzielnej pracy studenta</b>	<b>52</b> <i>(suma)</i>
21	<b>Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy</b> <i>(1 punkt ECTS=25-30 godzin obciążenia studenta)</i>	<b>2</b>
22	<b>Sumaryczne obciążenie pracą studenta</b>	<b>103</b>
23	<b>Punkty ECTS za moduł</b> <i>1 punkt ECTS=25-30 godzin obciążenia studenta</i>	<b>5</b>
24	<b>Nakład pracy związany z zajęciami o charakterze praktycznym</b> <i>Suma godzin związanych z zajęciami praktycznymi</i>	<b>43</b>
25	<b>Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym</b> <i>1 punkt ECTS=25-30 godzin obciążenia studenta</i>	<b>2</b>

### E. LITERATURA

Wykaz literatury	<ol style="list-style-type: none"><li>1. M. Ben-Ari, Podstawy programowania współbieżnego i rozproszonego, WNT</li><li>2. A.S. Tanenbaum, Rozproszone systemy operacyjne, PWN</li><li>3. Ch. Petzold, Programowanie Windows, Microsoft Press</li></ol>
Witryna WWW modułu/przedmiotu	<a href="http://achilles.tu.kielce.pl/Members/rtomaszewski/studia-zaoczne/materia142y-dydaktyczne-dla-studiow-zaocznych">http://achilles.tu.kielce.pl/Members/rtomaszewski/studia-zaoczne/materia142y-dydaktyczne-dla-studiow-zaocznych</a>