**Nazwa przedmiotu:**

Programowanie równoległe i rozproszone I

**Koordynator przedmiotu:**

dr Lucjan Stapp

**Status przedmiotu:**

Fakultatywny ograniczonego wyboru

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Wspólne

**Kod przedmiotu:**

**Semestr nominalny:**

6 / rok ak. 2009/2010

**Liczba punktów ECTS:**

4

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład: | 30h |
| Ćwiczenia: | 15h |
| Laboratorium: | 0h |
| Projekt: | 0h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

Algorytmy i struktury danych. Programowanie

**Limit liczby studentów:**

**Cel przedmiotu:**

Celem przedmiotu jest zaznajomienie studentów z możliwościami, jakie daje programowanie równoległe.

**Treści kształcenia:**

Po wprowadzeniu do programowania równoległego omawiane są następujące zagadnienia:
• metody synchronizacji dla komputerów z pamięcią wspólną
• podstawowe algorytmy dla systemów rozproszonych: wybór koordynatora, zakończenie działania , przydział zasobów
• problemy deadlocku i zagłodzenia
• podstawowe równoległe algorytmy numeryczne (działania na macierzach, rozwiązywanie układu równań liniowych) wraz z omówieniem zagadnień ich złożoności
Ćwiczenia poświęcone są tworzeniu algorytmów równoległych dla przykładowych zagadnień – głównie są to algorytmy wykorzystujące różne metody synchronizacji dla pamięci wspólnej.
Laboratoria służą praktycznemu zapoznaniu się studentów z zagadnieniami przedstawianymi na wykładzie środowisku linuxowym.

**Metody oceny:**

Wykład + ćwiczenia
test w połowie semestru 0 - 30 punktów
test na ostatnich zajęciach 0 - 80 punktów
aktywność na zajęciach -20 - +20 punktów
Laboratorium
9 zajęć 0 - 60 punktów
„mały” projekt 0 - 40 punktów
Ocena końcowa
if (wynik\_lab < 60 || wynik\_\_wyk < 70)
ocena = 2;
else
{
podstawa = wynik\_lab + wynik\_wyk - 130;
if (podstawa < 20) ocena = 3
elseif (podstawa < 40) ocena = 3.5
elseif (podstawa < 60 ) ocena = 4
elseif (podstawa < 80) ocena = 4.5
else ocena = 5;
}

**Egzamin:**

**Literatura:**

1. Parallel and Distributed Programming, F. Okulicka – Dłużewska, Lucjan Stapp, CEmS, Warszawa 2002
2. www.netlib.org

**Witryna www przedmiotu:**

**Uwagi:**

## Efekty przedmiotowe