**Nazwa przedmiotu:**

Systemy operacyjne (I)

**Koordynator przedmiotu:**

Tomasz Jordan KRUK

**Status przedmiotu:**

Fakultatywny ograniczonego wyboru

**Poziom kształcenia:**

Studia II stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Przedmioty techniczne - podstawowe

**Kod przedmiotu:**

SOI

**Semestr nominalny:**

1 / rok ak. 2018/2019

**Liczba punktów ECTS:**

5

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

- uczestnictwo w wykładach: 15 x 2 = 30 godzin
- uczestnictwo w zajęciach laboratoryjnych: 15 x 2 = 30 godzin
- przygotowanie do laboratoriów: 15 \* 2 = 30 godzin
- przygotowanie do kolokwium: 10 godzin
- egzamin i przygotowanie do egzaminu: 15 godzin

Razem: 30 + 30 + 30 + 10 + 15 = 115 godzin

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

- uczestnictwo w wykładach: 15 x 2 = 30 godzin
- uczestnictwo w zajęciach laboratoryjnych: 15 x 2 = 30 godzin

Razem 30 + 30 = 60 godzin co daje ok. 2,5 ECTS

**Język prowadzenia zajęć:**

polski

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

- uczestnictwo w zajęciach laboratoryjnych: 15 x 2 = 30 godzin
- przygotowanie do laboratoriów: 15 \* 2 = 30 godzin

Razem: 30 + 30 = 60 godzin co daje ok. 2,5 ECTS

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

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

**Wymagania wstępne:**

Umiejętność programowania w języku C.

**Limit liczby studentów:**

100

**Cel przedmiotu:**

Zapoznanie studentów z ideą, budową i własnościami systemów operacyjnych. Praktyczne wprowadzenie do programowania systemowego.

**Treści kształcenia:**

Wykład:
Wprowadzenie. Podstawowe definicje: system operacyjny, proces zasób. Funkcje jądra systemu. Historia rozwoju systemów operacyjnych. Tryby przetwarzania. Koncepcje i struktury systemów operacyjnych.
Programowanie w języku interpretera poleceń. Komendy powłoki. Przekierowanie wejścia/wyjścia. Metaznaki i rozpoznawanie wzorców. Zmienne powłoki i zasady substytucji.
Zarządzanie procesami. Stany procesów. Procesy a wątki. Architektury SMP.
Wzajemne wykluczenie i synchronizacja. Wyścigi i sekcje krytyczne. Komunikacja i synchronizacja międzyprocesowa. Klasyczne algorytmy synchronizacyjne.
Zarządzanie pamięcią. Organizacja pamięci. Adresowanie. Pamięć dzielona. Pamięć wirtualna. Algorytmy rozmieszczenia oraz wymiany. Segmentacja i stronicowanie.
Zarządzanie urządzeniami zewnętrznymi. Metody oprogramowywania wejścia wyjścia. Rozwiązania RAID.
Zarządzanie plikami. Struktura i organizacja systemy plików. Katalogi i pliki specjalne. Metody ochrony i kontroli dostępu.
Wirtualizacja. Hipernadzorca. Parawirtualizacja. Maszyny wirtualne.
Przetwarzanie rozproszone i rozwiązania klastrowe. Komunikacja międzyprocesowa w środowisku rozproszonym. Typy rozwiązań klastrowych. Zarządzanie procesami w środowisku rozproszonym.
Wiedza nabyta na wykładzie jest sprawdzana podczas kolokwium w trakcie semestru (Kol.) oraz egzaminu (E.).

Laboratorium:
Studenci realizują indywidualnie zadania dotyczące każdego z poniższych zagadnień:

Lab0. Wprowadzenie do środowiska laboratoryjnego
Lab1. Kompilacja jądra i dodanie wywołania systemowego
Lab2. Modyfikacja algorytmu szeregowania procesów
Lab3. Synchronizacja z wykorzystaniem semaforów
Lab4. Synchronizacja z wykorzystaniem monitorów
Lab5. Modyfikacja algorytmów zarządzania pamięcią operacyjną
Lab6. Tworzenie własnego systemu plików
Zadanie wprowadzające Lab0 wymaga zaliczenia ale nie jest punktowane. Zadania podzielone są na zadania krótkie (Lab0., Lab1., Lab5.) oraz długie (Lab2., Lab3., Lab4., Lab6.). Implementacja zadań długich obligatoryjnie poprzedzana jest sprawdzianami kontrolującymi przygotowanie merytoryczne (odpowiednio: We2, We3, We4, We6) oraz weryfikacją zaproponowanego zarysu rozwiązania.

**Metody oceny:**

Studenci na laboratorium realizują zadanie wprowadzające i 6 zadań punktowanych. Każde zadanie laboratoryjne oceniane jest niezależnie. Część zadań obejmuje wymóg zaliczenia wejściówki i zaprezentowania projektu koncepcji rozwiązania. W sumie z laboratorium można uzyskać 40 punktów. W trakcie semestru przeprowadzane jest kolokwium sprawdzające, a po semestrze egzamin. Z części wykładowej można uzyskać 60 punktów. W sumie można uzyskać 100 punktów.

**Egzamin:**

tak

**Literatura:**

1. Andrew S. Tanenbaum, Systemy operacyjne. Wydanie III. Wyd. Helion, 2010
2. William Stallings, Systemy operacyjne. Wydawnictwo Naukowe PWN, 2006

**Witryna www przedmiotu:**

https://usosweb.usos.pw.edu.pl/kontroler.php?\_action=katalog2/przedmioty/pokazPrzedmiot&prz\_kod=103A-INxxx-ISP-SOI; http://www.ia.pw.edu.pl/~tkruk/edu/soi.b/

**Uwagi:**

-

## Charakterystyki przedmiotowe

### Profil ogólnoakademicki - wiedza

**Charakterystyka WPO\_02:**

Student, który zaliczył przedmiot posiada wiedzę odnośnie metod komunikacji i synchronizacji międzyprocesowej.

Weryfikacja:

ocena wyników Kol i E

**Powiązane charakterystyki kierunkowe:** K\_W04

**Powiązane charakterystyki obszarowe:** I.P7S\_WG

### Profil ogólnoakademicki - umiejętności

**Charakterystyka UPO\_01:**

Student, który zaliczył przedmiot jest w stanie kompilować, modyfikować i rozszerzać system operacyjny.

Weryfikacja:

ocena wyników We2, We6 i Lab1., Lab2., Lab5. i Lab6.

**Powiązane charakterystyki kierunkowe:** K\_U11, K\_U13

**Powiązane charakterystyki obszarowe:** I.P7S\_UW, III.P7S\_UW.3.o

**Charakterystyka UPO\_02:**

Student, który zaliczył przedmiot jest w stanie prawidłowo wykorzystywać mechanizmy do synchronizacji i komunikacji międzyprocesowej.

Weryfikacja:

ocena wyników We3, We4 i Lab3., Lab4.

**Powiązane charakterystyki kierunkowe:** K\_U11, K\_U13

**Powiązane charakterystyki obszarowe:** I.P7S\_UW, III.P7S\_UW.3.o