**Nazwa przedmiotu:**

Systemy operacyjne

**Koordynator przedmiotu:**

Prof. nzw. dr hab. inż. Leszek Jan Opalski

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Wspólne

**Kod przedmiotu:**

**Semestr nominalny:**

3 / 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:  | 0h |
| Laboratorium:  | 15h |
| Projekt:  | 0h |
| Lekcje komputerowe:  | 0h |

**Wymagania wstępne:**

Podstawy programowania, Podstawy programowania strukturalnego

**Limit liczby studentów:**

**Cel przedmiotu:**

Wykład ma na celu zapoznanie studentów z podstawowymi pojęciami, koncepcjami, metodami i algorytmami występującymi we współczesnych systemach operacyjnych. Studenci powinni umieć m.in.:
- wyznaczyć średni czas oczekiwania i obiegu dla podanego zestawu zadań i różnych algorytmów szeregowania.
- zbadać (za pomocą algorytmu bankiera), czy zadany stan przydziałów zasobów jest bezpieczny i ew. znaleźć ciąg bezpieczny.
- określić średni czas dostępu do pamięci ze stronicowaniem/segmentacją.

Zajęcia laboratoryjne umożliwiają bliższe zapoznanie się z podstawowymi mechanizmami
systemowymi - na przykładzie systemu operacyjnego Linux. Studenci są przygotowywani do samodzielnego rozwiązywania niewielkich zadań programistycznych, wymagających:
- tworzenia i obsługi współbieżnych procesów oraz wątków, a także
- organizacji komunikacji pomiędzy procesami (pliki, pamięć współdzielona, łącza, kolejki komunikatów, gniazda)
- organizacji synchronizacji współpracujących zadań (wybór schematu: producent-konsument, czytelnicy-pisarze, biesiadujący filozofowie, wybór środków synchronizacji: semafory, muteksy, zmienne warunkowe).

**Treści kształcenia:**

Wykład:
1.  Wprowadzenie. Zadania i właściwości systemów operacyjnych. Struktury systemów komputerowych i systemów operacyjnych. Struktura i interfejs jądra systemu.
2.  Procesy i wątki. Koncepcje procesu, procesu lekkiego i wątku; cykl życia i interfejs programisty. Realizacja współbieżności procesów i wątków, szeregowanie, wywłaszczanie. Sygnały i wyjątki oraz ich obsłu­ga.
3.  Komunikacja międzyprocesowa i synchronizacja. Komunikacja między procesami: pamięć dzielona, system komunikatów; synchronizacja współpracujących procesów i wątków: Metody realizacji i interfejs programisty. Problem zak­leszczania: model i metody postępowania.
4.  Zarządzanie pamięcią pomocniczą. Plik, organizacja systemu plików. Operacje na systemie plików i interfejs programisty. Dostęp i ochrona plików, tryby dostępu do plików.
5.  Zarządzanie systemem wejścia/wyjścia. Urządzenia znakowe i blokowe; tryby dostępu. Moduły obsługi urządzeń wejścia/wyjścia.
6.  Komunikacja sieciowa. Model komunikacji OSI. Interfejs gniazd w programowaniu połączeniowych i bezpołączeniowych aplikacji sieciowych. Problemy komunikacji i synchronizacji aplikacji sieciowych. Przykłady usług sieciowych.
7.  Ochrona i bezpieczeństwo systemów komputerowych. Cele ochrony. Uwierzytelnianie, kontrola dostępu do zasobów, integralność, niezaprzeczalność, poufność– koncepcje i realizacje. Standardy oceny bezpieczeństwa.

Laboratorium:
1.  Procesy i sygnały.
2.  Komunikacja i synchronizacja międzyprocesowa (cz. 1 i 2)
3.  Operacje wejścia/wyjścia
4.  Komunikacja sieciowa (interfejs gniazd)
5.  Wątki POSIX – tworzenie, synchronizacja

**Metody oceny:**

Ocena z przedmiotu wynika z sumy punktów uzyskanych przez studenta w laboratorium (maksymalnie 6\*7p) oraz za sprawdziany wykładowe (58p). Pozytywna ocena końcowa wymaga uzyskania co najmniej 50p, w tym 20p. z laboratorium.
Laboratorium – zasady szczegółowe:

Do dyspozycji studenta są zajęcia wprowadzające i 6 terminów zajęć ocenianych.
W każdym terminie zajęć ocenianych można uzyskać 7p  (2p – wejściówka, 5p – rozwiązania zadań programistycznych), stąd łącznie 42p.
Rozwiązania wszystkich zadań podlegających ocenie muszą być przesłane prowadzącemu zajęcia w czasie zajęć z własnego konta studenta na serwerze Wydziale MiNI, bądź być przekazane (w czasie zajęć) prowadzącemu na dyskietce. Student ma przekazać prowadzącemu kody źródłowe wszystkich opracowanych programów oraz skrypt powłoki, bądź plik Makefile, który można użyć do uzyskania programu wykonywalnego.W treści zadania laboratoryjnego może być zawarte wymaganie testowania opracowanego programu. W tym przypadku student przekazuje prowadzącemu dodatkowo pliki danych testujących, a wyniki testów przedstawia w krótkim sprawozdaniu. Wszystkie wymienione powyżej pliki tekstowe muszą zawierać nazwisko i grupę studencką autora.
Zajęcia odbywają się (bez przerw) w s. 221/222 wg harmonogramu (dostępnego pod URL: http://alpha.mini.pw.edu.pl/~opalskil/SOP/harmonogram.html) uzgodnionego z reprezentacją studentów w ciągu pierwszych 2 tygodni semestru.
Nie ma dodatkowego terminu zajęć laboratoryjnych na odrabianie nieobecności, czy poprawianie oceny.
Jedna usprawiedliwiona nieobecność powoduje zastąpienie brakującej oceny przez średnią pozostałych.

Sprawdziany wykładowe – zasady szczegółowe:

W semestrze odbędą się 2 sprawdziany wykładowe w terminach określonych w harmonogramie.
Usprawiedliwiona nieobecność daje prawo do odbycia sprawdzianu w dodatkowym terminie (terminie sprawdzianu poprawkowego). Obowiązuje ten sam zakres materiału co w pierwotnym terminie. Student musi zgłosić wykładowcy zamiar przystąpienia do tego sprawdzianu najpóźniej jedną dobę przed tym terminem
Osoby, które uzyskały co najmniej 20p z laboratorium - mogą odbyć pisemny sprawdzian poprawkowy w terminie dodatkowym (patrz harmonogram). Zakres tego sprawdzianu (za 58p) obejmuje całość materiału wykładowego. Wynik tego sprawdzianu zastępuje sumę punktów z dwóch sprawdzianów semestralnych (uwaga: wynik sprawdzianu poprawkowego może być mniejszy od sumy punktów ze sprawdzianów w semestrze). Student musi zgłosić wykładowcy zamiar przystąpienia do sprawdzianu poprawkowego najpóźniej jedną dobę przed tym sprawdzianem.

Uwagi ogólne:

Wszystkie oceniane prace muszą być wykonywane samodzielnie. Niesamodzielność pracy, bądź korzystanie przez studenta z niedozwolonych materiałów powoduje uzyskanie z danej pracy/sprawdzianu 0pt.
W czasie wykonywania ocenianych prac pisemnych (jak wejściówki laboratoryjne i sprawdziany wykładowe) nie można korzystać z żadnych materiałów pisanych, nagrań dźwiękowych, środków komunikacji elektronicznej.
W czasie realizacji zadań laboratoryjnych (po wejściówce) można korzystać z własnych materiałów, rozwiązań zadań przykładowych, dokumentacji systemowej oraz Internetu jednak tak, by oceniana praca była dziełem samodzielnym.
Wszelkie usprawiedliwienia nieobecności powinny być zgłoszone prowadzącemu przedmiot przed zakończeniem semestru, nie później niż w terminie następnego najbliższego wykładu od ustania przyczyny nieobecności.

**Egzamin:**

**Literatura:**

1.  A. Silberschatz,  P.B. Galvin, G. Gagne, Podstawy systemów operacyjnych, wyd. 6 zm. i rozsz., WNT 2005
2.  W. Stallings, Systemy Operacyjne, Wyd. Robomatic, 2004
3.  U. Vahalia, Jądro systemu UNIX. Nowe horyzonty, WNT 2001
4.  R.W. Stevens, Programowanie w środowisku systemu UNIX, WNT 2002

**Witryna www przedmiotu:**

**Uwagi:**

## Efekty przedmiotowe