



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Systemy operacyjne [S1Inf1>SOP]

Przedmiot

Kierunek studiów
Informatyka

Rok/Semestr
1/2

Studia w zakresie (specjalność)
–

Profil studiów
ogólnoakademicki

Poziom studiów
pierwszego stopnia

Język oferowanego przedmiotu
polski

Forma studiów
stacjonarne

Wymagalność
obligatoryjny

Liczba godzin

Wykład
30

Laboratorium
30

Inne
0

Ćwiczenia
0

Projekty/seminaria
0

Liczba punktów ECTS

5,00

Koordynatorzy

dr inż. Dariusz Wawrzyniak
dariusz.wawrzyniak@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający ten moduł powinien posiadać podstawową wiedzę z zakresu funkcjonowania komputera i programowania imperatywnego. Powinien posiadać umiejętność rozwiązywania podstawowych problemów z zakresu implementacji i oceny kosztu działania prostych algorytmów oraz umiejętność pozyskiwania informacji ze wskazanych źródeł. Powinien również rozumieć konieczność poszerzania swoich kompetencji i wykazywać gotowość do podjęcia współpracy w ramach zespołu. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.

Cel przedmiotu

1. Przekazanie studentom podstawowej wiedzy na temat funkcjonowania systemów operacyjnych w zakresie zarządzania zasobami systemu komputerowego.
2. Zapoznanie studentów z koncepcją planowania przydziału procesora, zarządzania pamięcią i urządzeniami wejścia-wyjścia oraz organizacją systemu plików.
3. Rozwinięcie u studentów umiejętności rozwiązywania prostych problemów zarządzania systemem komputerowym, w tym ochrony zasobów systemu i informacji.
4. Kształtowanie u studentów umiejętności organizacji przetwarzania z uwzględnieniem wydajności i optymalnego wykorzystania zasobów systemu.

Przedmiotowe efekty uczenia się

Wiedza:

1. Zna wybrane zagadnienia dotyczące systemów operacyjnych.
2. Zna zagadnienia z zakresu algorytmów i struktur danych oraz podstawy teorii złożoności obliczeniowej.
3. Zna zagadnienia z zakresu stosowalności algorytmów optymalizacji kombinatorycznej.
4. Zna zasady programowania strukturalnego i obiektowego.

Umiejętności:

1. Potrafi projektować i tworzyć oprogramowanie z wykorzystaniem usług systemu operacyjnego zgodnie z zadaną specyfikacją, używając właściwych metod, technik i narzędzi.
2. Potrafi dokonać analizy funkcjonalności i analizy wymagań systemów informatycznych w kontekście wykorzystania usług systemu operacyjnego.
3. Potrafi pozyskiwać informacje z literatury, baz danych oraz innych właściwie dobranych źródeł, także w języku angielskim.

Kompetencje społeczne:

1. Rozumie potrzebę uczenia się przez całe życie i podnoszenia swoich kompetencji.
2. Potrafi współdziałać i pracować w grupie, przyjmując w niej różne role.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- 1.
2. W zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 -
 - dyskusję i odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach.
 -
- 3.
4. W zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 -
 - ocenę umiejętności związanych z realizacją ćwiczeń laboratoryjnych.
 -
- 5.

Ocena podsumowująca:

- 1.
2. W zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 -
 - ocenę wiedzy i umiejętności wykazanych na egzaminie końcowym w formie testu składającego się z około 10 zadań oraz pytań otwartych lub zamkniętych o zróżnicowanej punktacji, z możliwością uzyskania w sumie około 100 punktów, z progiem na ocenę pozytywną od 50% punktów;
 -
 - omówienie/przedyskutowanie wyników testu na żądanie studenta.
 -
- 3.

4. W zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:
- - ocenę wiedzy i umiejętności związanych z treściami przekazywanymi na zajęciach poprzez kolokwium końcowe lub kolokwia cząstkowe;
 -
 - ocenę postępu realizacji zadań ćwiczeniowych.
 -

Aktywność podczas zajęć może być premiowana dodatkowymi punktami, w szczególności za:

-
- omówienie dodatkowych aspektów zagadnienia,
-
- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanej problemu,
-
- uwagi prowadzące do udoskonalenia materiałów dydaktycznych lub procesu dydaktycznego.
-

Treści programowe

Program wykładu obejmuje następujące zagadnienia:

- 1.
2. Definicja i funkcje systemu operacyjnego, klasyfikacja systemów operacyjnych, struktura oprogramowania systemowego i jego związek ze sprzętem, zasada działania jądra systemu operacyjnego.
- 3.
4. System plików:
 - 1.
 2. organizacja logiczna: definicja pliku i jego atrybuty, metody dostępu do pliku, interfejs operacji plikowych, logiczna struktura katalogów.
 - 3.
 4. organizacja fizyczna: przydział bloków dyskowych (ciągły, łańcuchowy i indeksowy), zarządzanie wolną przestrzenią (wektor bitowy, lista łączona, grupowanie, zliczanie), implementacja katalogu (lista liniowa, tablica haszowa, struktura indeksowe); realizacja operacji plikowych (buforowa pamięć podręczna, problem integralność, współbieżny dostęp do pliku).
 - 5.
 6. przykłady implementacji: CP/M, DOS, ISO 9660, Unix, NTFS.
 - 7.
- 5.
6. Ogólna koncepcja zarządzania zasobami systemu komputerowego.
- 7.
8. Zarządzanie procesorem: planowanie przydziału czasu procesora, kryteria uszeregowania; algorytmy planowania przydziału procesora.
- 9.
10. Zarządzanie pamięcią operacyjną: ewolucja organizacji pamięci, przydział pamięci, tworzenie obrazu procesu w pamięci, stronicowanie i segmentacja.
- 11.
12. Pamięć wirtualna: błąd braku strony, efektywność działania, problem zastępowania stron, algorytmy wymiany stron
- 13.
14. Zarządzanie urządzeniami wejścia/wyjścia: klasyfikacja urządzeń wejścia/wyjścia, struktura mechanizmu we/wy, interakcja jednostki centralnej z urządzeniami wejścia wyjścia, buforowanie i spooling.
- 15.

Zajęcia laboratoryjne prowadzone są w formie piętnastu dwugodzinnych zajęć w laboratorium komputerowym. Ćwiczenia realizowane są indywidualnie przez studentów. Tematyka dotyczy użytkownika uniksopodobnego systemu operacyjnego z elementami programowania w oparciu o usługi jądra systemu operacyjnego (wywołania systemowe) i obejmuje następujące zagadnienia:

- 1.

2. Podstawowe pojęcia: logowanie, interpreter poleceń, terminal jako urządzenie, użytkownicy, grupy, hierarchiczna struktura katalogów, korzystanie z pomocy.
- 3.
4. Obsługa systemu plików: podstawowe operacje na plikach i katalogach (kopiowanie, relokowanie, tworzenie dowiązań), wzorce uogólniające, przeszukiwanie (find).
- 5.
6. Prawa dostępu: interpretacja praw dostępu do plików zwykłych i katalogów, sposoby zmiany praw dostępu.
- 7.
8. Obsługa procesów: tworzenie wykazu procesów, unicestwienie, zmiana priorytetu, sposoby uruchamiania procesów (sekwencyjne, współbieżne, w tle, warunkowe, z przekierowaniem strumieni, potokowe).
- 9.
10. Filtry: head, tail, more, grep, cut, tr, sort, uniq, sed itp.
- 11.
12. Programy użytkowe: cat, cmp, comm, wc, edytor vi itp.
- 13.
14. Shell: zmienne lokalne i środowiskowe, aliasy, funkcje.
- 15.
16. Skrypty: konstrukcje warunkowe, pętle, instrukcja wyboru, obsługa zmiennych specjalnych (w tym parametrów pozycyjnych).
- 17.
18. Programowa obsługa plików: tworzenie i usuwanie plików zwykłych, realizacja dostępu do plików (otwieranie, zapis, odczyt).
- 19.

Tematyka zajęć

Program wykładu obejmuje następujące zagadnienia:

- 1.
2. Definicja i funkcje systemu operacyjnego, klasyfikacja systemów operacyjnych, struktura oprogramowania systemowego i jego związek ze sprzętem, zasada działania jądra systemu operacyjnego.
- 3.
4. System plików:
 - 1.
 2. organizacja logiczna: definicja pliku i jego atrybuty, metody dostępu do pliku, interfejs operacji plikowych, logiczna struktura katalogów.
 - 3.
 4. organizacja fizyczna: przydział bloków dyskowych (ciągły, łańcuchowy i indeksowy), zarządzanie wolną przestrzenią (wektor bitowy, lista łączona, grupowanie, zliczanie), implementacja katalogu (lista liniowa, tablica haszowa, struktura indeksowe); realizacja operacji plikowych (buforowa pamięć podręczna, problem integralność, współbieżny dostęp do pliku).
 - 5.
 6. przykłady implementacji: CP/M, DOS, ISO 9660, Unix, NTFS.
 - 7.
- 5.
6. Ogólna koncepcja zarządzania zasobami systemu komputerowego.
- 7.
8. Zarządzanie procesorem: planowanie przydziału czasu procesora, kryteria uszeregowania; algorytmy planowania przydziału procesora.
- 9.
10. Zarządzanie pamięcią operacyjną: ewolucja organizacji pamięci, przydział pamięci, tworzenie obrazu procesu w pamięci, stronicowanie i segmentacja.
- 11.
12. Pamięć wirtualna: błąd braku strony, efektywność działania, problem zastępowania stron, algorytmy wymiany stron
- 13.

14. Zarządzanie urządzeniami wejścia/wyjścia: klasyfikacja urządzeń wejścia/wyjścia, struktura mechanizmu we/wy, interakcja jednostki centralnej z urządzeniami wejścia wyjścia, buforowanie i spooling.
- 15.

Zajęcia laboratoryjne prowadzone są w formie piętnastu dwugodzinnych zajęć w laboratorium komputerowym. Ćwiczenia realizowane są indywidualnie przez studentów. Tematyka dotyczy użytkownika uniksopodobnego systemu operacyjnego z elementami programowania w oparciu o usługi jądra systemu operacyjnego (wywołania systemowe) i obejmuje następujące zagadnienia:

- 1.
2. Podstawowe pojęcia: logowanie, interpreter poleceń, terminal jako urządzenie, użytkownicy, grupy, hierarchiczna struktura katalogów, korzystanie z pomocy.
- 3.
4. Obsługa systemu plików: podstawowe operacje na plikach i katalogach (kopiowanie, relokowanie, tworzenie dowiązań), wzorce uogólniające, przeszukiwanie (find).
- 5.
6. Prawa dostępu: interpretacja praw dostępu do plików zwykłych i katalogów, sposoby zmiany praw dostępu.
- 7.
8. Obsługa procesów: tworzenie wykazu procesów, unicestwianie, zmiana priorytetu, sposoby uruchamiania procesów (sekwencyjne, współbieżne, w tle, warunkowe, z przekierowaniem strumieni, potokowe).
- 9.
10. Filtry: head, tail, more, grep, cut, tr, sort, uniq, sed itp.
- 11.
12. Programy użytkowe: cat, cmp, comm, wc, edytor vi itp.
- 13.
14. Shell: zmienne lokalne i środowiskowe, aliasy, funkcje.
- 15.
16. Skrypty: konstrukcje warunkowe, pętle, instrukcja wyboru, obsługa zmiennych specjalnych (w tym parametrów pozycyjnych).
- 17.
18. Programowa obsługa plików: tworzenie i usuwanie plików zwykłych, realizacja dostępu do plików (otwieranie, zapis, odczyt).
- 19.

Metody dydaktyczne

Wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, rozwiązywanie zadań.

Ćwiczenia laboratoryjne: wykonywanie eksperymentów, rozwiązywanie zadań, dyskusja, praca w zespole.

Literatura

Podstawowa:

- 1.
2. A. Silberschatz, P. B. Galvin, G. Gagne, Podstawy systemów operacyjnych, PWN, W-wa, 2021.
- 3.
4. W. Stallings, Systemy operacyjne. Architektura, funkcjonowanie i projektowanie. wyd. 9, Helion, 2018.
- 5.
6. A.S. Tanenbaum, H. Bos, Systemy operacyjne. wyd. 4, Helion, 2015.
- 7.
8. C. Sobaniec, System operacyjny Linux — przewodnik użytkownika, Nakom, Poznań, 2002.
- 9.

Uzupełniająca:

- 1.
2. G. Nutt, Operating Systems. A Modern Perspective, Addison Wesley Longman, Inc., 2002.
- 3.

4. B. Goodheart, J. Cox, Sekrety magicznego ogrodu. UNIX System V Wersja 4 od środka, WNT, W-wa, 2001.
- 5.
6. U. Vahalia, Jądro systemu UNIX, WNT, W-wa, 2001.
- 7.
8. P. Yosifovich, A. Ionescu, M.E. Russinovich, D.A. Solomon, Windows od środka, wyd. 7, Helion, 2018.
- 9.
10. J. Marczyński, UNIX użytkowanie i administrowanie, wyd. 2, Helion, Gliwice, 2000.
- 11.
12. A.S. Tanenbaum, Strukturalna organizacja systemów komputerowych. Wydanie V, Helion, Gliwice, 2006.
- 13.
14. D. Wawrzyniak, Systemy operacyjne i sprzęt informatyczny, W: Informatyka gospodarcza, A. Gąsioriewicz, K. Rostek, J. Zawila-Niedźwiecki (red.), Wydawnictwo C.H. Beck, Warszawa, 2010.
- 15.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	125	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	62	2,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	63	2,50