

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Systemy operacyjne
Nazwa w języku angielskim:		Operating Systems
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		obowiązkowy
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia, jednolitych magisterskich):		pierwszego stopnia
Rok studiów:	pierwszy	
Semestr:		drugi
Liczba punktów ECTS:	3	
Imię i nazwisko koordynatora przedmiotu:		Prof. dr hab. inż. Franciszek Seredyński
Imię i nazwisko prowadzących zajęcia:		Prof. dr hab. inż. Franciszek Seredyński, dr Piotr Świtalski, mgr Michał Seredyński
Założenia i cele przedmiotu:		Celem przedmiotu jest zaznajomienie studentów z organizacją i mechanizmami systemów operacyjnych oraz zagadnieniami programowania z użyciem języków skryptowych. W szczególności zostaną przedstawione takie zagadnienia jak: • struktura systemów operacyjnych, • mechanizmami zarządzania procesorem i pamięcią w systemach komputerowych, • mechanizmy synchronizacji procesów.
Symbol efektu	Efekty uczenia się	Symbol efektu kierunkowego
	WIEDZA Student zna i rozumie:	
W_01	podstawowe struktury systemów operacyjnych	K_W05, K_W14
W_02	mechanizmy zarządzania procesami, ich synchronizacji oraz algorytmy szeregowania procesów przez system operacyjny	K_W07, K_W14
W_03	mechanizmy zarządzania pamięcią operacyjną i sposoby organizacji tej pamięci	K_W14
	UMIEJĘTNOŚCI Student potrafi:	
U_01	sprawnie wyszukiwać w literaturze informacje związane z systemami operacyjnymi oraz rozwiązywać problemy w trakcie posługiwania się systemem operacyjnym poprzez umiejętność czytania dokumentacji tego systemu	K_U01, K_U26
U_02	zarządzać systemem operacyjnym oraz wykonywać typowe zadania użytkownika, w tym tworzenie skryptów powłoki	K_U22, K_U26
U_03	rozwiązać proste problemy związane z instalacją i konfiguracją systemu operacyjnego	K_U26

Forma i typy zajęć:	Studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (30 godz.); Studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)
Wymagania wstępne i dodatkowe:	
Warunkiem uczestnictwa w zajęciach jest znajomość zagadnień związanych z architekturą systemów komputerowych oraz podstawami programowania lub znajomość literatury obowiązującej w tych przedmiotach.	
Treści modułu kształcenia:	
Treści dotyczące wykładu:	
1. Architektura i oprogramowanie komputerów – rys historyczny. Koncepcja maszyny Johna von Neumanna. Komputery głównego szeregu (komputery mainframe). Superkomputery i systemy wieloprocesorowe. Sieci komputerowe i Internet. Systemy mobilnej komunikacji. Systemy sensorowe i Internet rzeczy. Chmury obliczeniowe. 2. Struktury systemów komputerowych cz. 1. Architektura systemu komputerowego. Działanie systemu komputerowego. Ogólne funkcje architektury przerwań. Obsługa przerwań. Przerwania wejścia/wyjścia. Struktura DMA. Struktura pamięci. Pamięć cache. 3. Struktury systemów komputerowych cz. 2. Struktura pamięci. Szeregowanie dostępu do danych na dysku. Hierarchia pamięci. Buforowanie podręczne. Dualny tryb operacji. Ochrona pamięci. Ochrona procesora. 4. Struktury systemów operacyjnych cz. 1. Składowe systemu operacyjnego. Zarządzanie procesami. Zarządzanie pamięcią operacyjną. Zarządzanie plikami. Zarządzanie systemem wejścia/wyjścia. Zarządzanie pamięcią pomocniczą. System interpretacji poleceń. Usługi systemu operacyjnego. Wywołania systemowe. Rodzaje wywołań systemowych. 5. Struktury systemów operacyjnych cz. 2. Działanie wczesnych systemów jednozadaniowych. Struktura systemu UNIX. Modele komunikacji procesów. Standard POSIX. Programy systemowe. Podejście warstwowe. Maszyny wirtualne. Projektowanie i implementacja systemu. Generowanie systemu. 6. Procesy cz. 1. Koncepcja procesu. Stan procesu. Blok kontrolny procesu. Przełączanie kontekstu. Kolejki planowania procesów. Migracja procesów między kolejkami systemu. Planisci. Tworzenie i kończenie procesów. 7. Procesy cz. 2. Współdziałające procesy. Problem producenta-konsumenta. Wątki. Struktura wątku. Komunikacja międzyprocesowa. Buforowanie. Zdalne wywoływanie procedur. 8. Planowanie przydziału procesora. Pojęcia podstawowe. Cykl faz procesora i wejścia-wyjścia. Planista i ekspedytor. Kryteria planowania. Planowanie metodą FCFS. Planowanie metodą „najpierw najkrótsze zadanie”. Planowanie priorytetowe. Planowanie rotacyjne. Kolejki wielopoziomowe. Planowanie zadań dla wielu procesorów. Ocena algorytmów planowania. 9. Synchronizowanie procesów. Podstawy. Zagadnienie producenta-konsumenta. Problem sekcji krytycznej. Rozwiązania wieloprocesowe. Sprzętowe środki synchronizacji. Semaforey i monitory. Zakleszczenia i głodzenie. Problem ograniczonego buforowania. Klasyczne problemy synchronizacji i ich rozwiązania. 10. Zarządzanie pamięcią cz. 1. Podstawy. Wiązanie adresów, ładowanie i konsolidacja. Konsolidacja dynamiczna. Logiczna i fizyczna przestrzeń adresowa. Nakładki. Wymiana procesów. Przydział	

ciągły. Fragmentacja. Stronicowanie. Budowa tablicy stron. Sprzęt stronicujący z buforami TLB. Efektywny czas dostępu do pamięci.

11. **Zarządzanie pamięcią cz. 2.** Ochrona pamięci. Stronicowanie wielopoziomowe. Odwrócona tablica stron. Strony dzielone. Ochrona pamięci. Segmentacja. Segmentacja ze stronicowaniem. Stronicowanie w architekturach IA-32 oraz x86-64.
12. **Pamięć wirtualna cz. 1.** Podstawy. Koncepcja pamięci wirtualnej. Stronicowanie na żądanie. Zastępowanie stron. Algorytmy zastępowania stron: FIFO, optymalny, LRU. Algorytmy przybliżające metodę LRU.
13. **Pamięć wirtualna cz. 2.** Algorytmy zliczające. Algorytm przydziału ramek. Szamotanie. Unikanie szamotania. Model zbioru roboczego.
14. **Interfejs systemu plików.** Budowa systemu plików. Pojęcie pliku. Operacje plikowe. Tablica otwartych plików. Struktura pliku. Metody dostępu. Struktura katalogowa. Ochrona systemu plików. Semantyka spójności.
15. **Systemy operacyjne w systemach chmurowych i Internecie Rzeczy (IoT).** Obliczenia w chmurze. Chmurowe systemy operacyjne. Internet rzeczy. Systemy operacyjne w IoT.

Treści dotyczące laboratorium:

1. **Wprowadzenie do systemów operacyjnych** – rodzaje powłok, składnia poleceń, podstawowe polecenia Linuksa.
2. **Używanie poleceń powłoki.** Używanie zmiennych powłoki. Tworzenie własnego środowiska powłoki.
3. **Zarządzanie procesami.** Procesy, demony i sygnały – zarządzanie wykonaniem w środowisku SO.
4. **System plików.** Obsługa plików i katalogów, użytkownicy, grupy i prawa dostępu.
5. **Praca z plikami tekstowymi.** Filtrowanie strumienia tekstu i wyrażenia regularne.
6. **Wyszukiwanie i przetwarzanie wzorców.** Polecenie awk.
7. **Wyrażenia regularne.** Klasy znaków. Grupy. Kwantyfikatory. Wyrażenia regularne i grep.
8. **Skrypty powłoki cz. 1.** Cytowania. Zmienne w skryptach. Funkcje. Polecenie test. Konstrukcje if/fi.
9. **Skrypty powłoki cz. 2.** Wykonywanie operacji arytmetycznych. Modyfikatory zmiennych. Struktura case. Struktura for...do.
10. **Skrypty powłoki cz. 3.** Struktura while...do. Struktura until...do. Polecenie shift. Polecenia break oraz true. Polecenie continue.
11. **Instalacja systemu Linux.** Wstęp do administracji systemem.
12. **Zarządzanie dyskami i systemami plików.** Użytkownicy i grupy.
13. **Linux w kontenerach.**
14. **Cykliczne uruchamianie programów oraz system kontroli wersji git.**
15. **PowerShell.**

Literatura podstawowa:

1. Silberschatz A., Galvin P. B., Gagne G.: Podstawy systemów operacyjnych, tom I, wydanie X, PWN, 2021.
2. Tanenbaum A. S., Bos H.: Systemy operacyjne. wydanie V, Helion, 2024.

Literatura dodatkowa:

1. Stallings W.: Systemy operacyjne. Architektura, funkcjonowanie i projektowanie. Wydanie IX, Helion 2018.
2. Negus Ch., Linux. Biblia. Wydanie X, Helion, 2021.

Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany jest technikami multimedialnymi. Ćwiczenia laboratoryjne – zajęcia praktyczne z wykorzystaniem wybranych narzędzi programistycznych. Na stronie internetowej prowadzącego zamieszczane są materiały z problemami i zadaniami laboratoryjnymi.

Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Symbol efektu	Metody weryfikacji efektów uczenia się
W_01	Efekt realizowany w ramach kolokwium w trakcie trwania semestru oraz na egzaminie w formie pisemnej metodą pytań wielokrotnego wyboru. Przykładowe pytanie: „Czym są rozkazy uprzywilejowane – wybierz poprawne odpowiedzi z listy: a) są to rozkazy wywołań systemowych, b) rozkazy programu wykonywane w pierwszej kolejności, c) rozkazy programu potencjalnie niebezpieczne dla systemu operacyjnego, d) rozkazy pojawiające się bardzo rzadko”.
W_02	Efekt realizowany w ramach kolokwium w trakcie trwania semestru oraz na egzaminie w formie pisemnej metodą pytań wielokrotnego wyboru. Przykładowe pytanie: „Wybierz własności algorytmu FCFS uszeregowania procesów z zadanego zbioru odpowiedzi”.
W_03	Efekt realizowany w ramach kolokwium w trakcie trwania semestru oraz na egzaminie w formie pisemnej metodą pytań wielokrotnego wyboru. Przykładowe pytanie: „Wybierz cechy opisujące koncepcję i właściwości pamięci stronicowanej”.
U_01	Efekt realizowany w trakcie zajęć laboratoryjnych. Sprawdzane będą umiejętności korzystania z poleceń powłoki bash np. man, whereis, help oraz interpretowania zwracanych zapisów. Dodatkowo student będzie w stanie wyszukać dokumentację powłoki danego systemu operacyjnego i sprawnie z niej korzystać. Przykładowe zadanie: „Znajdź informacje jak wyświetlić w powłoce bash status wyjścia ostatnio wykonanego polecenia”.
U_02	Efekt realizowany w trakcie zajęć laboratoryjnych. Sprawdzane będą umiejętności tworzenia skryptów powłoki w oparciu o poznane struktury języka programowania. Przykładowe zadanie: „Przygotuj skrypt, który nada prawo wykonywania plikom bez takiego prawa, które znajdują się na liście argumentów skryptu”.
U_03	Efekt realizowany w trakcie zajęć laboratoryjnych. Student będzie realizował zadanie związane z instalacją i konfiguracją systemu operacyjnego. Przykładowe zadanie: „Przeprowadź instalację systemu operacyjnego Fedora na podstawie zamontowanego nośnika instalacyjnego na maszynie wirtualnej i przeprowadź partycjonowanie dysku w następujący sposób: /boot (1024 MB), / (6 GB), /var (2 GB) i /home (2 GB)”.

Forma i warunki zaliczenia:

Ocena z przedmiotu składa się z dwóch ocen częściowych:

- oceny z zajęć laboratoryjnych,
- oceny z egzaminu w formie pisemnej.

Na ocenę z zajęć laboratoryjnych składają się oceny częściowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać sumarycznie 40 pkt. Zaliczenie zajęć laboratoryjnych możliwe jest po uzyskaniu co najmniej 51% liczby punktów z tej formy zaliczenia.

Warunkiem przystąpienia studenta do kolokwium egzaminacyjnego jest uzyskanie pozytywnej oceny z laboratorium.

Zaliczenie wykładu w formie egzaminu nastąpi w przypadku uzyskania co najmniej 51% liczby z puli 60 punktów przewidzianej dla tej formy zaliczenia.

Sumarycznie można zdobyć 100 punktów.

Ocena końcowa, w zależności od sumy uzyskanych punktów (maksymalnie 100 punktów) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	30 godz.
Przygotowanie się do kolokwium śródsesestralnego i kolokwium egzaminacyjnego	5 godz.
Przygotowanie się do zajęć laboratoryjnych	7 godz.
Udział w konsultacjach z przedmiotu	3 godz.
Sumaryczne obciążenie pracą studenta	75 godz.
Punkty ECTS za przedmiot	3

Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach laboratoryjnych	15 godz.
Przygotowanie się do kolokwium egzaminacyjnego	21 godz.
Przygotowanie się do zajęć laboratoryjnych	21 godz.
Udział w konsultacjach z przedmiotu	3 godz.
Sumaryczne obciążenie pracą studenta	75 godz.
Punkty ECTS za przedmiot	3