

KARTA PRZEDMIOTU**I. Dane podstawowe**

Nazwa przedmiotu	Systemy operacyjne
Nazwa przedmiotu w języku angielskim	Operating systems
Kierunek studiów	Informatyka
Poziom studiów (I, II, jednolite magisterskie)	I stopnia
Forma studiów (stacjonarne, niestacjonarne)	Stacjonarne
Dyscyplina	Informatyka
Język wykładowy	polski

Koordinator przedmiotu	Dr Viktor Melnyk, prof. KUL
------------------------	-----------------------------

Forma zajęć (<i>katalog zamknięty ze słownika</i>)	Liczba godzin	semestr	Punkty ECTS
wykład	15	IV	3
konwersatorium			
ćwiczenia			
laboratorium	30	IV	
warsztaty			
seminarium			
proseminarium			
lektorat			
praktyki			
zajęcia terenowe			
pracownia dyplomowa			
translatorium			
wizyta studyjna			

Wymagania wstępne	W1 - znajomością treści z informatyki objętych programem szkoły średniej. W2 - znajomość podstawowych zagadnień z architektury komputerów. W3 - znajomość z podstawami programowania. W4 - dobra znajomość obsługi komputera.
-------------------	--

II. Cele kształcenia dla przedmiotu

C_01 - zaznajomienie słuchaczy z podstawowymi pojęciami oraz ideami wykorzystywanymi w systemach operacyjnych historycznych i współczesnych
C_02 - przedstawienie konkretnych rozwiązań zastosowanych w systemach rodzin Unix oraz Windows.

III. Efekty uczenia się dla przedmiotu wraz z odniesieniem do efektów kierunkowych

Symbol	Opis efektu przedmiotowego	Odniesienie do efektu kierunkowego
WIEDZA		
W_01	Student ma teoretyczną wiedzę z zakresu systemów operacyjnych, stosowanych struktur danych i algorytmów	K_W01 K_W04
W_02	Student zna funkcje podstawowych katalogów	K_W01 K_W04
W_03	Student zna nazewnictwo urządzeń oraz reprezentację ich jako plików	K_W01 K_W04
W_04	Student zna podstawowe polecenia powłoki	K_W01 K_W04
W_05	Student zna podstawową konfigurację usług DNS, SMTP, POP3, WWW	K_W01 K_W04
UMIEJĘTNOŚCI		
U_01	Umiejętność korzystania z narzędzi programisty w systemie Unix/Linux	K_U01 K_U02
U_02	Student potrafi zastosować wybrane funkcje systemowe API systemu operacyjnego Unix/Linux	K_U01 K_U02 K_U04
U_03	Student ma podstawowe umiejętności tworzenia i zarządzania procesami w systemie Unix/Linux	K_U01 K_U02
U_04	Student potrafi zainstalować wybraną dystrybucję systemu Unix/Linux	K_U01 K_U02 K_U04
U_05	Student potrafi zarządzać użytkownikami i grupami zabezpieczeń, jak również odczytywać i nadawać im uprawnienia do plików	K_U01 K_U02 K_U17
U_06	Student potrafi zamontować i odmontować urządzenie w drzewie katalogów	K_U01 K_U02
U_07	Student potrafi dokonać konfiguracji podstawowych ustawień systemowych z poziomu wiersza poleceń	K_U01 K_U02
U_08	Student potrafi uruchamiać podstawowe usługi dostępne w systemie Unix/Linux	K_U01 K_U02 K_U04
U_09	Student potrafi pisać proste skrypty w powłoce systemowej	K_U01 K_U02 K_U04 K_U19
KOMPETENCJE SPOŁECZNE		
K_01	Umiejętnie rozwiązywać złożone problemy, z którymi może spotkać się w życiu, stosując poznane zasady działania systemów operacyjnych, obiektywnie oceniając uzyskane wyniki	K_K01 K_K02
K_02	Postępować ze standardami etycznymi obowiązującymi w branży IT	K_K01 K_K04
K_03	Pracować efektywnie, sprawnie, zespołowo i indywidualnie, umiejętnie oceniając priorytety w realizacji projektu	K_K01 K_K02

IV. Opis przedmiotu/ treści programowe**WYKŁAD**

1. Istota, rola i zadania systemu operacyjnego. Rodzaje systemów operacyjnych. Podstawy działania. Struktura systemu. Historyczny rys rozwoju systemów operacyjnych - od systemów wsadowych do interakcyjnych.
2. Zasoby, procesy i wątki. Właściwości i pożądane cechy systemów operacyjnych. Przetwarzanie współbieżne.
3. Planowanie przydziału procesora. Algorytmy planowania i kryteria oceny. Implementacja algorytmów planowania procesów i wątków.

4. Systemowe mechanizmy synchronizacji procesów. Komunikacja między procesami - wzajemne wyłączenie, synchronizacja i blokada. Semaforzy - zasada działania, implementacja. Zastosowanie semaforów w komunikacji międzyprocesowej.
5. Hierarchia pamięci. Zarządzanie pamięcią operacyjną. Sterowanie pamięcią, przydział pamięci.
6. Pamięć wirtualna. Sposoby zamieszczenia stron w pamięci operacyjnej.
7. System plików. Atrybuty pliku. Podstawowe operacje na plikach. Operacje na katalogu. Organizacja logiczna systemu plików. Organizacja fizyczna systemu plików.
8. Mechanizmy wejścia/wyjścia. Rodzaje urządzeń wejścia-wyjścia. Struktura mechanizmu wejścia-wyjścia. Buforowanie.

LABORATORIUM

1. Instalacja systemu operacyjnego.
2. Wprowadzenie oraz obsługa systemu plików w Linux'ie.
3. Linux – konsola. Operacje na katalogach i plikach. Konta użytkowników. Montowanie i odmontowywanie urządzeń.
4. Procesy, zmienne, programy, pliki, standardowe wyjście i wejście.
5. Filtry, strumienie standardowe oraz przetwarzanie potokowe.
6. Tworzenie skryptów powłoki systemu operacyjnego Linux.
7. Obsługa edytora vi.
8. Obsługa edytora joe.
9. Windows: Power shell, pliki batch.

V. Metody realizacji i weryfikacji efektów uczenia się

Symbol efektu	Metody dydaktyczne <i>(lista wyboru)</i>	Metody weryfikacji <i>(lista wyboru)</i>	Sposoby dokumentacji <i>(lista wyboru)</i>
WIEDZA			
W_01	Wykład konwencjonalny	Egzamin / Zaliczenie pisemne	Karta egzaminacyjna / Karta zaliczeniowa
W_02, W_03, W_04, W_05	Wykład konwencjonalny, Ćwiczenia laboratoryjne	Egzamin / Zaliczenie pisemne, Sprawdzenie umiejętności praktycznych, Obserwacja	Karta egzaminacyjna / Karta zaliczeniowa, Karta oceny / Raport z obserwacji, Protokół / Wydruk / Plik sprawozdania
UMIĘTNOŚCI			
U_01 - U_09	Ćwiczenia laboratoryjne, Ćwiczenia praktyczne design thinking	Sprawdzenie umiejętności praktycznych, Obserwacja	Karta oceny / Raport z obserwacji Protokół / Wydruk / Plik sprawozdania
KOMPETENCJE SPOŁECZNE			
K_01, K_02	Wykład konwencjonalny, Ćwiczenia laboratoryjne design thinking	Egzamin / Zaliczenie pisemne, Sprawdzenie umiejętności praktycznych, Obserwacja	Karta egzaminacyjna / Karta zaliczeniowa, Karta oceny / Raport z obserwacji, Protokół / Wydruk / Plik sprawozdania
K_03	Ćwiczenia laboratoryjne design thinking	Sprawdzenie umiejętności praktycznych, Obserwacja	Karta oceny / Raport z obserwacji, Protokół / Wydruk / Plik sprawozdania

VI. Kryteria oceny, wagi...

Warunkiem zaliczenia ćwiczeń jest obecność studenta na zajęciach dydaktycznych, wykonanie ćwiczeń i otrzymywanie ocen, uzyskanie pozytywnej oceny za odpowiedzi na pytania kontrolne w każdym ćwiczeniu.

Oceny za wykonanie ćwiczeń - 70%, obecność - 30%

Egzamin (dla osób, które zaliczyły ćwiczenia) polega na przeprowadzeniu testu z wiedzy przekazanej na wykładzie. Ocena egzaminacyjna jest formowana na podstawie dwóch składowych:
70 % - odpowiedzi pisemne na zadania testowe i odpowiedzi ustne w razie wątpliwości,
30% - ocena otrzymana z ćwiczeń.

Ocenianie jest wykonywane według następującej skali:

90 – 100% - bardzo dobry (5.0),

80 – 89% - plus dobry (4.5),

70 – 79% - dobry (4.0),

60 – 69% - plus dostateczny (3.5),

50 – 59% - dostateczny (3.0),

poniżej 50% - niedostateczny (2.0).

Szczegółowe zasady oceniania są podawane studentom z każdą edycją przedmiotu.

VII. Obciążenie pracą studenta

Forma aktywności studenta	Liczba godzin
Liczba godzin kontaktowych z nauczycielem	45
Liczba godzin indywidualnej pracy studenta	60

VIII. Literatura

Literatura podstawowa
<ol style="list-style-type: none"> 1. Andrew S. Tanenbaum: Systemy operacyjne. Wydanie IV. Helion, 2015. 2. William Stallings: Systemy operacyjne. Architektura, funkcjonowanie i projektowanie. Wydanie IX. Helion, 2018 3. Shechovtsov V. Operating systems : Textbook for high-school students. BHV, 2005. 4. Abraham Silberschatz, Peter B. Galvin, Greg Gagne: Podstawy systemów operacyjnych. WNT, Warszawa 2006
Literatura uzupełniająca
<ol style="list-style-type: none"> 1. M.J. Bach: Budowa systemu operacyjnego Unix, WNT, Warszawa 1994 2. T.W. Ogletree: Windows XP PL. Księga eksperta, Helion, Gliwice 2002 3. U. Vahalia: Jądro systemu Unix, WNT, Warszawa 2000 4. D.A. Solomon: Inside Windows NT, Microsoft Press, 1998 5. B. Goodheart, J. Cox: Sekrety magicznego ogrodu. UNIX® System V Wersja 4 od środka. WNT, Warszawa 2001. 6. U. Vahalia: Jądro systemu UNIX®. Nowe horyzonty. WNT, W-wa, 2001. 7. D. A. Solomon, M. E. Russinovich: Microsoft Windows® 2000 od środka, Helion, 2003. 8. R. Lowe: Kernel Linux. Przewodnik programisty, Helion, 2004 9. Daniel P. Bovet, Marco Cesati , Understanding the Linux Kernel, 3rd Edition. O'Reilly Media (November 2005): 944 Pages. 10. Jonathan Levin, Mac OS X and iOS Internals: To the Apple's Core. Wiley (2012), 864 pages