

# PAŃSTWOWA WYŻSZA SZKOŁA ZAWODOWA W NOWYM SĄCZU

## KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2016/2017

Instytut Techniczny

Kierunek studiów: Informatyka

Profil: Praktyczny

Forma studiów: Stacjonarne

Kod kierunku: 11.3

Stopień studiów: I

Specjalności: Informatyka stosowana

### 1 PRZEDMIOT

NAZWA PRZEDMIOTU	Programowanie systemowe
KOD PRZEDMIOTU	IT 11.3 PIS C14 16/17
KATEGORIA PRZEDMIOTU	Przedmioty specjalnościowe
LICZBA PUNKTÓW ECTS	2
SEMESTRY	7

### 2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

SEMESTR	WYKŁAD	ĆWICZENIA	LABORATORIUM	PROJEKT	SEMINARIUM
7				30	

### 3 CELE PRZEDMIOTU

**Cel 1** Student potrafi tworzyć aplikacje oraz rozwiązania dotyczące tworzenia, szeregowania i zarządzania procesami i wątkami, zarządzania pamięcią, komunikacji między procesami, obsługi urządzeń i systemów plików.



## 4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

- a Matematyka dyskretna
- b Podstawy programowania C++, JAVA, SQL oraz PHP
- c Technologie sieciowe
- d Sieci lokalne oraz rozległe
- e Podstawy elektroniki i miernictwa

## 5 EFEKTY KSZTAŁCENIA

- EK1** Wiedza: Ma szczegółową wiedzę z zakresu programowania, ma wiedzę na temat sposobu implementacji aplikacji z użyciem języków programowania C++, JAVA, SQL oraz PHP, zna podstawowe paradygmaty programowania.
- EK2** Wiedza: Ma wiedzę w zakresie programowania współbieżnego i rozproszonego, zna techniki synchronizacji procesów oraz podstawowe algorytmy rozproszone.
- EK3** Umiejętności: Potrafi ocenić przydatność i sposób funkcjonowania, istniejące rozwiązania elementów informatycznych, możliwość ich zastosowania dla konkretnego systemu lub sieci informatycznej.
- EK4** Umiejętności: Potrafi dokonać analizy krytycznej wyników funkcjonalnego i strukturalnego testowania systemu informatycznego.

## 6 TREŚCI PROGRAMOWE

### PROJEKT

LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
P1	Procesy i wątki. Budowa procesów i wątków. Modele wielowątkowości	4
P2	Tworzenie modułów w Linuxie	4
P3	Metody "odplukiwania" jądra Linuxa I	4
P4	Sygnały	4
P5	Planista i zarządzanie pamięcią w systemie Linux	2
P6	Przerwania	2
P7	Dodawanie wywołania systemowego w jądrze Linuxa	2
P8	Metody synchronizacji jądra Linuxa	4
P9	Metody "odplukiwania" jądra Linuxa II	4
	RAZEM	<b>30</b>

## 7 METODY DYDAKTYCZNE

- M1 Ćwiczenia projektowe
- M2 Prezentacje multimedialne
- M3 Słowne objaśnienie



## 8 OBCIĄŻENIE PRACĄ STUDENTA

FORMA AKTYWNOŚCI	ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI
<b>Godziny kontaktowe z nauczycielem akademickim, w tym:</b>	
Godziny wynikające z planu studiów	30
Konsultacje przedmiotowe	0
Egzaminy i zaliczenia w sesji	1
<b>Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym:</b>	
Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	5
Opracowanie wyników	4
Przygotowanie raportu, projektu, prezentacji, dyskusji	5
Instalacji oprogramowania	5
<b>SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA</b>	<b>50</b>
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	2

## 9 SPOSOBY OCENY

### OCENA FORMUJĄCA

**F1** Sprawozdanie z ćwiczenia laboratoryjnego

**F2** Aktywność na zajęciach

**F3** Projekt indywidualny

### OCENA AKTYWNOŚCI BEZ UDZIAŁU NAUCZYCIELA AKADEMICKIEGO

**1** Ćwiczenie praktyczne

**2** Projekt indywidualny

### KRYTERIA OCENY

EFEKT KSZTAŁCENIA 1		MIEJSCE WERYFIKACJI	OPIS WERYFIKACJI EK 1
NA OCENĘ 3	Student posiada szczegółową wiedzę z zakresu programowania, ma wiedzę na temat sposobu implementacji aplikacji z użyciem języków programowania C++, JAVA, SQL oraz PHP, zna podstawowe paradygmaty programowania. Robi dużo błędów.	projekt	prezentacja sprawozdania projektu
NA OCENĘ 4	Student posiada szczegółową wiedzę z zakresu programowania, ma wiedzę na temat sposobu implementacji aplikacji z użyciem języków programowania C++, JAVA, SQL oraz PHP, zna podstawowe paradygmaty programowania. Nie robi poważnych błędów.		



NA OCENĘ 5	Student posiada szczegółową wiedzę z zakresu programowania, ma wiedzę na temat sposobu implementacji aplikacji z użyciem języków programowania C++, JAVA, SQL oraz PHP, zna podstawowe paradygmaty programowania. Nie robi wcale błędów.		
EFEKT KSZTAŁCENIA 2		MIEJSCE WERYFIKACJI	OPIS WERYFIKACJI EK 2
NA OCENĘ 3	Student posiada wiedzę w zakresie programowania współbieżnego i rozproszonego, zna techniki synchronizacji procesów oraz podstawowe algorytmy rozproszone. Robi dużo błędów.	projekt	prezentacja sprawozdania projektu
NA OCENĘ 4	Student posiada wiedzę w zakresie programowania współbieżnego i rozproszonego, zna techniki synchronizacji procesów oraz podstawowe algorytmy rozproszone. Nie robi poważnych błędów.		
NA OCENĘ 5	Student posiada wiedzę w zakresie programowania współbieżnego i rozproszonego, zna techniki synchronizacji procesów oraz podstawowe algorytmy rozproszone. Nie robi wcale błędów.		
EFEKT KSZTAŁCENIA 3		MIEJSCE WERYFIKACJI	OPIS WERYFIKACJI EK 3
NA OCENĘ 3	Potrafi ocenić przydatność i sposób funkcjonowania, istniejące rozwiązania elementów informatycznych, możliwość ich zastosowania dla konkretnego systemu lub sieci informatycznej. Robi dużo błędów.	projekt	prezentacja sprawozdania projektu
NA OCENĘ 4	Potrafi ocenić przydatność i sposób funkcjonowania, istniejące rozwiązania elementów informatycznych, możliwość ich zastosowania dla konkretnego systemu lub sieci informatycznej. Nie robi poważnych błędów.		
NA OCENĘ 5	Potrafi ocenić przydatność i sposób funkcjonowania, istniejące rozwiązania elementów informatycznych, możliwość ich zastosowania dla konkretnego systemu lub sieci informatycznej. Nie robi wcale błędów.		
EFEKT KSZTAŁCENIA 4		MIEJSCE WERYFIKACJI	OPIS WERYFIKACJI EK 4
NA OCENĘ 3	Potrafi dokonać analizy krytycznej wyników funkcjonalnego i strukturalnego testowania systemu informatycznego. Robi dużo błędów.	projekt	prezentacja sprawozdania projektu
NA OCENĘ 4	Potrafi dokonać analizy krytycznej wyników funkcjonalnego i strukturalnego testowania systemu informatycznego. Nie robi poważnych błędów.		



NA OCENĘ 5	Potrafi dokonać analizy krytycznej wyników funkcjonalnego i strukturalnego testowania systemu informatycznego. Nie robi wcale błędów.		
------------------	---	--	--

#### OCENA DO INDEKSU (OCENA PODSUMOWUJĄCA)

Ocena końcowa wystawiana jako średnia z ocen projektów

#### WARUNKI ZALICZENIA PRZEDMIOTU

- a Ocena kolokwium teorii
- b Oceny z prezentacji sprawozdań projektów

### 10 MACIERZ REALIZACJI PRZEDMIOTU

EFEKTY KSZTAŁCENIA DLA PRZEDMIOTU	ODNIESIENIE DO EFEKTÓW KIERUNKOWYCH	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	METODY DYDAKTYCZNE
EK1	INFP_UB04, INFP_W11, INFP_UB01, INFP_W09	Cel1	P1, P3, P5, P6	M1, M2, M3
EK2	INFP_UB04, INFP_W11, INFP_UB01, INFP_W09	Cel1	P3, P4, P7	M1, M2
EK3	INFP_UB04, INFP_W11, INFP_W09	Cel1	P4, P7, P8, P9	M2, M3
EK4	INFP_UB04, INFP_W11	Cel1	P2, P4, P5, P6, P7, P9	M1, M2, M3

### 11 WYKAZ LITERATURY

#### LITERATURA PODSTAWOWA:

- [1] 1.Abraham Silberschatz, James L. Peterson — *Podstawy systemów operacyjnych*, Warszawa, 2005, WNT
- [2] 2.William Stallings — *Operating Systems. Internals and Design Principles*, USA, 2005, Prentice Hall
- [3] 3.William Stallings — *Systemy operacyjne*, Warszawa, 2006, PWN
- [4] 4.Andrew S. Tanenbaum, Albert S. Woodhull — *Operating Systems. Design and Implementation*, Amazon.com, 2006, Prentice-Hall International, Inc.
- [5] 5.H. M. Deitel, P. J. Deitel, D.R. — *Choffnes, Operating Systems*, Amazon.com, 2004, Prentice Hall
- [6] 6.Uresh Vahalia — *Jądro systemu Unix..*, Warszawa, 2001, WNT



- [7] 7. Marshall Kirk McKusick, George V. — *Neville-Neil, The Design and Implementation of the FreeBSD Operating System*, Amazon.com, 2005, Addison Wesley
- [8] 8. Marshall Kirk McKusick — *The Design and Implementation of the FreeBSD Operating System*, Amazon.com, 2010, Addison Wesley

#### LITERATURA UZUPEŁNIAJĄCA:

- [1] 1. Daniel P. Bovet, Marco Cesati — *Understanding the Linux Kernel*, USA, 2006, O'Reilly
- [2] 2. Robert Love — *Linux Kernel Development*, USA, 2005, Noval Press
- [3] 3. Claudia Salzberg Rodriguez, Gordon Fisher, Steven Smolski — *The Linux Kernel Primer. A Top Down Approach for x86 and PowerPC Architecture*, USA, 2006, Prentice Hall Professional Technical Reference
- [4] 4. Steve Best — *Linux Debugging and Performance Tunning. Tips and Techniques*, USA, 2005, Prentice Hall Professional Technical Reference
- [5] 5. Christian Benvenuti — *Understanding Linux Network Internals*, USA, 2005, O'Reilly

## 12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

#### OSOBA ODPOWIEDZIALNA ZA KARTĘ

prof. dr hab. inż. Oleksandr Petrov (kontakt: asp1951@gmail.com)

#### OSOBY PROWADZĄCE PRZEDMIOT

prof. dr hab. inż. Oleksandr Petrov (kontakt: asp1951@gmail.com)

## 13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejscowość, data)	(odpowiedzialny za przedmiot)	(kierownik zakładu)	(dyrektor instytutu)
---------------------	-------------------------------	---------------------	----------------------

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....