



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie sieciowe

Przedmiot

Kierunek studiów

Bioinformatyka

Studia w zakresie (specjalność)

Poziom studiów

pierwszego stopnia

Forma studiów

stacjonarne

Rok/semestr

2/4

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

30

Laboratoria

30

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

4

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Michał Sajkowski, doc. PP

email: Michal.Sajkowski@put.poznan.pl

tel. 61 6653062

Instytut Informatyki

ul. Piotrowo 2, 60-965 Poznań

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Andrzej Stroiński

email: andrzej.stroinski@cs.put.poznan.pl

tel. 61 6652371

Instytut Informatyki

ul. Piotrowo 2, 60-965 Poznań

Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu organizacji systemów komputerowych, algorytmów i struktur danych oraz systemów operacyjnych.

Powinien posiadać umiejętność formułowania algorytmów i ich programowania z użyciem przynajmniej jednego z popularnych narzędzi oraz umiejętność pozyskiwania informacji ze wskazanych źródeł.

Powinien również rozumieć konieczność poszerzania swoich kompetencji i mieć 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 z sieci komputerowych, w zakresie użytkowania, konfigurowania i projektowania lokalnych i rozległych sieci komputerowych oraz wiedzy specjalistycznej



w zakresie programowania sieci komputerowych.

2. Rozwijanie u studentów umiejętności rozwiązywania prostych problemów powstałych przy programowaniu sieciowym oraz użytkowaniu i konfigurowaniu sieci komputerowych.
3. Kształtowanie u studentów umiejętności pracy zespołowej, zwłaszcza przy konfigurowaniu, projektowaniu i programowaniu rozwiązań technicznych stosowanych w sieciach komputerowych.

Przedmiotowe efekty uczenia się

Wiedza

1. zna wybrane zagadnienia dotyczące technologii sieciowych i programowania sieciowego [K1st_W16]
2. ma podstawową wiedzę o cyklu życia systemów informatycznych [K1st_W22].

Umiejętności

1. potrafi przygotować, w języku ojczystym i angielskim, dobrze udokumentowane opracowanie oraz prezentację ustną dotyczącą zagadnień z zakresu z zakresu programowania sieciowego [K1st_U08]
2. potrafi dokonać analizy funkcjonalności i analizy wymagań systemów informatycznych [K1st_U14]
3. projektuje i tworzy oprogramowanie komputerowe zgodnie z zadaną specyfikacją, używając właściwych metod, technik i narzędzi [K1st_U15]

Kompetencje społeczne

1. rozumie potrzebę uczenia się przez całe życie i podnoszenia swoich kompetencji [K1st_K01]
2. potrafi współdziałać i pracować w grupie, przyjmując w niej różne role [K1st_K02]
3. potrafi odpowiednio określić priorytety służące realizacji zadania określonego przez siebie lub innych [K1st_K03]

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenianie ciągłe, na każdych zajęciach (odpowiedzi ustne) – premiowanie przyrostu umiejętności posługiwania się poznanymi zasadami i metodami

b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę przygotowania studenta do poszczególnych sesji zajęć laboratoryjnych (sprawdzian wejściowy) oraz ocenę umiejętności związanych z realizacją ćwiczeń laboratoryjnych
- ocenę sprawozdania przygotowywanego częściowo w trakcie zajęć, a częściowo po ich zakończeniu; ocena ta obejmuje także umiejętność pracy w zespole
- ocenę wiedzy i umiejętności związanych z realizacją projektu programistycznego, realizowanego przez studenta jako praca domowa
- ocenę wiedzy i umiejętności związanych z realizacją zadań laboratoryjnych poprzez 1 do 2 kolokwiów w semestrze.



Ocena podsumowująca

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę wiedzy i umiejętności wykazanych na kolokwium pisemnym o charakterze problemowym, składającym się z 3 do 5 zadań, albo od 10 do 15 pytań testowych. Aby zaliczyć kolokwium i uzyskać ocenę 3.0, student musi uzyskać co najmniej 50% maksymalnej liczby punktów. W trakcie kolokwium student nie może korzystać z materiałów dydaktycznych

b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- zestawienie ocen wystawionych w trakcie semestru w postaci średniej ważonej, 1/2 oceny będzie stanowiła ocena z projektu programistycznego, 1/2 oceny będzie stanowiła ocena z kolokwiów.

Treści programowe

W ramach wykładu studenci poznają następujące zagadnienia:

- 1) Podstawy sieci komputerowych (rys historyczny, motywacja, cechy wymagane od sieci, architektura sieci - OSI i TCP/IP, topologie sieci, typy sieci, urządzenia sieciowe)
- 2) Internet (struktura, adresowanie, protokoły IP, UDP i TCP, standardy)
- 3) Programowanie sieciowe – wprowadzenie
- 4) Model klient-serwer i budowa oprogramowania
- 5) Interfejs programowy dla protokołów komunikacyjnych
- 6) Interfejs gniazd
- 7) Algorytmy i problemy realizacji oprogramowania klienta
- 8) Algorytmy i problemy realizacji serwerów
- 9) Przykłady programów klienckich i programów serwerów

W ramach ćwiczeń laboratoryjnych studenci poznają następujące zagadnienia:

- 1) adresacja IPv4
- 2) konfiguracja systemu Linux do pracy w sieci IP
- 3) model warstwowy ISO/OSI
- 4) tworzenie pakietów
- 5) protokoły warstw niższych
- 6) wprowadzenie do interfejsu gniazd sieciowych
- 7) model serwera z użyciem fork()
- 8) model serwera z użyciem wątków
- 9) model serwera w modelach zdarzeniowych
- 10) biblioteka Twisted tworzenia serwerów w Python
- 11) biblioteka gevent oparta na zielonych wątkach dla języka Python
- 12) programowanie serwerów warstwy sesji
- 13) programowanie sieciowe usług w modelu sieciowych kolejek komunikatów



Metody dydaktyczne

1. wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy.
2. ćwiczenia laboratoryjne: rozwiązywanie zadań, ćwiczenia praktyczne z wykorzystaniem urządzeń sieciowych, dyskusja, praca w zespole, pokaz multimedialny, demonstracja, projekt programistyczny realizowany w domu i weryfikowane na ćwiczeniach laboratoryjnych.

Literatura

Podstawowa

1. UNIX - programowanie usług sieciowych. 1 API: gniazda i XTI, W.R. Stevens, WNT, 2002.
2. Sieci komputerowe TCP/IP, Tom 3: Programowanie w trybie klient-serwer. Wersja BSD, D.E. Comer, D.L. Stevens, WNT, Warszawa, 1997
3. Sieci komputerowe, wyd. V, A.S. Tanenbaum, D.J. Wetherall, Helion, Gliwice, 2012
4. Sieci komputerowe, wyd. 7, J.F. Kurose, K.W. Ross, Helion, Gliwice, 2019
5. TCP/IP Protocol Suite, 4th ed., B.A. Forouzan, McGraw-Hill Education, New York 2009
6. Data Communications and Networking, 5th ed., B.A. Forouzan, McGraw-Hill Education, New York 2012

Uzupełniająca

4. Diagnozowanie i utrzymywanie sieci. Księga eksperta, J. Scott Haugdahl, Helion, Gliwice, 2000.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	100	4,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	62	2,5
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/zaliczenia pisemnego, wykonanie projektu programistycznego) ¹	38	1,5

¹ niepotrzebne skreślić lub dopisać inne czynności