

| KARTA OPISU MODUŁU KSZTAŁCENIA | | |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nazwa modułu/przedmiotu Programowanie sieciowe | | Kod 1010512311010510015 |
| Kierunek studiów Informatyka | Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki | Rok / Semestr 1 / 1 |
| Ścieżka obieralności/specjalność Systemy rozproszone | Przedmiot oferowany w języku: polski | Kurs (obligatoryjny/obieralny) obligatoryjny |
| Stopień studiów: II stopień | Forma studiów (stacjonarna/niestacjonarna) stacjonarna | |
| Godziny Wykłady: 30 Ćwiczenia: - Laboratoria: 30 Projekty/seminaria: - | | Liczba punktów 4 |
| Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) kierunkowy | | (ogólnouczelniany, z innego kierunku) ogólnouczelniany |
| Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne nauki techniczne | | Podział ECTS (liczba i %) 4 100% 4 100% |
| Odpowiedzialny za przedmiot / wykładowca: dr inż. Michał Kalewski email: Michal.Kalewski@cs.put.poznan.pl tel. 61 6652370 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań | | |
| Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych: | | |
| 1 | Wiedza: | Efekty kształcenia ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_W1-2, K_W4, K_W6-15, weryfikowane w procesie rekrutacji na studia 2 stopnia ? efekty te prezentowane są w serwisie internetowym wydziału www.fc.put.poznan.pl Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu systemów operacyjnych, programowania współbieżnego oraz sieci komputerowych. |
| 2 | Umiejętności: | Efekty kształcenia ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_U1-2, K_U4, K_U7-8, K_U14-20, K_U22-23, K_U26, weryfikowane w procesie rekrutacji na studia 2 stopnia ? efekty te prezentowane są w serwisie internetowym wydziału www.fc.put.poznan.pl |
| 3 | Kompetencje społeczne | Efekty kształcenia ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_K1-9, weryfikowane w procesie rekrutacji na studia 2 stopnia ? efekty te prezentowane są w serwisie internetowym wydziału www.fc.put.poznan.pl 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 wiedzy z dziedziny programowania sieciowego w zakresie prezentacji teoretycznych i praktycznych aspektów implementacji programów i systemów wykorzystujących współczesne sieci komputerowe. 2. Zapoznanie studentów z problematyką rozwiązań technicznych stosowanych obecnie przy implementacji aplikacji sieciowych, a szczególności z problematyką architektury i konstrukcji serwerów i klientów sieciowych oraz programowalnych sieci komputerowych. 3. Rozwijanie u studentów umiejętności rozwiązywania problemów, z jakimi spotyka się programista aplikacji sieciowych. 4. Zdobycie przez studentów umiejętności wykorzystania i konfigurowania istniejących protokołów i narzędzi systemowych do projektowania i tworzenia oprogramowania sieciowego. | | |
| Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia | | |
| Wiedza: | | |

| |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none">1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie algorytmów i złożoności, architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, języków i paradygmatów programowania, grafiki i komunikacji człowiek-komputer, sztucznej inteligencji, baz danych, inżynierii oprogramowania, wspomagania decyzji oraz systemów wbudowanych - [K_W4]2. ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki - [K_W5]3. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce i w wybranych pokrewnych dyscyplinach naukowych - [K_W6]4. ma podstawową wiedzę o cyklu życia systemów informatycznych sprzętowych lub programowych - [K_W7]5. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z wybranego obszaru informatyki - [K_W8] |
| Umiejętności: |
| <ol style="list-style-type: none">1. potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł (w języku ojczystym i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie - [K_U1]2. potrafi określić kierunki dalszego uczenia się i zrealizować proces samokształcenia - [K_U5]3. potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne - [K_U9]4. potrafi - przy formułowaniu i rozwiązywaniu zadań inżynierskich - integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne - [K_U10]5. potrafi formułować i testować hipotezy związane z problemami inżynierskimi i prostymi problemami badawczymi - [K_U12]6. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K_U13]7. potrafi zaproponować ulepszenia (usprawnienia) istniejących rozwiązań technicznych - [K_U21] |
| Kompetencje społeczne: |
| <ol style="list-style-type: none">1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]2. zna możliwości dalszego dokształcania się (III stopnia, studia podyplomowe, kursy i egzaminy przeprowadzane przez uczelnie, firmy i organizacje zawodowe) - [K_K3]3. zna przykłady i rozumie przyczyny wadliwie działających systemów informatycznych, które doprowadziły do poważnych strat finansowych, społecznych lub też do poważnej utraty zdrowia, a nawet życia - [K_K4]4. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]5. potrafi myśleć i działać w sposób przedsiębiorczy - [K_K8] |

Sposoby sprawdzenia efektów kształcenia

Ocena podsumowująca:

- a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 - ocenę wiedzy i umiejętności wykazanych na zaliczeniu pisemnym obejmującym około 5 zagadnień omawianych w ramach wykładów o różnej wartości punktowej wg następującej skali:
 - 0 - 50% ocena: 2,0
 - 51 - 60% ocena: 3,0
 - 61 - 70% ocena: 3,5
 - 71 - 80% ocena: 4,0
 - 81 - 90% ocena: 4,5
 - 91 - 100% ocena: 5,0
 - omówienie wyników zaliczenia
- b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 - ocenę umiejętności związanych z realizacją ćwiczeń laboratoryjnych,
 - ocenianie ciągle, na każdych zajęciach (odpowiedzi ustne) - premiowanie przyrostu umiejętności posługiwania się poznanymi zasadami i metodami,
 - ocenę wiedzy i umiejętności związanych z realizacją zadań projektowych poprzez 1 kolokwium w semestrze;uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:
 - omówienia dodatkowych aspektów zagadnienia,
 - efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,
 - umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium,
 - uwagi związane z udoskonaleniem materiałów dydaktycznych,
 - wskazywanie trudności percepcyjnych studentów, umożliwiające bieżące doskonalenia procesu dydaktycznego.

Treści programowe

Program wykładu obejmuje następujące zagadnienia:

- Wprowadzenie: przypomnienie materiału dotyczącego modelu warstwowego sieci komputerowych oraz podstawowych datagramowych (UDP) i strumieniowych (TCP) gniazd sieciowych protokołu IPv4.
- Bufory nadawcze i odbiorcze gniazd sieciowych oraz metody i narzędzia kapsułkowania danych aplikacyjnych w transmisjach strumieniowych.
- Opcje systemowe i funkcje pomocnicze dla komunikacji internetowej interfejsu gniazd sieciowych oraz obsługa systemu nazw domenowych DNS.
- Dostęp programistyczny do ramek warstwy łącza danych (biblioteki libpcap oraz libnet) oraz pakietów sieciowych w jądrze systemu operacyjnego (biblioteki libnetfilter).
- Surowe gniazda sieciowe dla protokołów IPv4 i IPv6 oraz gniazda typu SOCK_PACKET i PF_PACKET.
- Gniazda sieciowe protokołu SCTP (Stream Control Transmission Protocol) oraz implementacje komunikacji wielostrumieniowych.
- Datagramowe i strumieniowe gniazda sieciowe protokołu IPv6; współdziałanie procesów wykorzystujących protokoły IPv4 i IPv6; konstrukcje dualnych serwerów sieciowych.
- Mechanizmy i architektury współbieżnych serwerów sieciowych: nieblokujące funkcje sieciowe; multipleksacja wejścia-wyjścia (funkcje systemowe select, poll, epoll i kqueue); funkcje sieciowe wyzwalane sygnałami systemowymi; współbieżność z zastosowaniem procesów potomnych oraz wątków; pule procesów potomnych oraz wątków i mechanizmy ich aktywacji do obsługi komunikacji sieciowej; mechanizm SO_REUSEPORT.
- Metody i mechanizmy detekcji rozłączeń w strumieniowej komunikacji internetowej.
- Implementacje internetowej komunikacji rozgłoszeniowej oraz multicastowej z użyciem interfejsu gniazd sieciowych.
- Interfejsy programistyczne do obsługi tablic routingu i pamięci podręcznej protokołu ARP oraz gniazda sieciowe zarządzania kluczami (typu PF_KEY).
- Interfejs gniazd sieciowych w systemach operacyjnych Windows (winsock), Windows Phone, Android i iOS; wykorzystanie interfejsu gniazd sieciowych do realizacji komunikacji w bezprzewodowych sieciach mobilnych.
- Mechanizmy obsługi komunikacji sieciowej w aplikacjach z graficznym interfejsem użytkownika.
- Implementacje komunikacji internetowej z zastosowaniem protokołów warstwy aplikacji (biblioteka libcurl).
- Wprowadzenie do programowalnych sieci komputerowych SDN (Software-Defined Networking), protokołu OpenFlow oraz emulatora Mininet; wykorzystanie protokołu OpenFlow do implementacji kontrolerów urządzeń sieciowych w sieciach SDN.

Program laboratorium obejmuje następujące zagadnienia:

- Wykorzystanie bibliotek libpcap, libnet oraz libnetfilter do implementacji programów przechwytyjących ramki warstwy łącza danych oraz pakiety sieciowe w jądrze systemu operacyjnego.
- Implementacje prostych narzędzi trasowania w sieciach IPv4 i IPv6 z użyciem surowych gniazd sieciowych (ICMPv4 i ICMPv6).
- Implementacje aplikacji klienta i serwera wykorzystujących gniazda sieciowe protokołu SCTP i połączenia wielostrumieniowe oraz testy efektywnościowe takich połączeń.
- Implementacje aplikacji klientów i serwerów z użyciem datagramowych i strumieniowych gniazd sieciowych protokołu IPv6; modyfikacje aplikacji serwerów do rozwiązań dualnych (IPv4/IPv6).
- Implementacja wybranych mechanizmów i architektur serwerów współbieżnych i ich porównawcze testy efektywnościowe.
- Implementacje komunikacji rozgłoszeniowych i multicastowych w sieciach internetowych.
- Implementacje programów manipulujących tablicami routingu i pamięcią podręczną protokołu ARP.
- Implementacje aplikacji klientów sieciowych dla wybranych systemów operacyjnych urządzeń mobilnych, obsługa graficznych interfejsów użytkownika oraz komunikacja w bezprzewodowych sieciach mobilnych.
- Wykorzystanie biblioteki libcurl do implementacji komunikacji internetowej z zastosowaniem protokołów warstwy aplikacji.
- Obsługa emulatora Mininet oraz implementacje kontrolerów urządzeń sieciowych w sieciach SDN z zastosowaniem protokołu OpenFlow.

Metody dydaktyczne:

1. wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, rozwiązywanie zadań, pokaz multimedialny
2. ćwiczenia: rozwiązywanie zadań, ćwiczenia praktyczne, dyskusja, praca w zespole, pokaz multimedialny, studium przypadków

Literatura podstawowa:

1. UNIX - programowanie usług sieciowych. 1, API: gniazda i XTI, Stevens W. R., Wydawnictwa Naukowo-Techniczne, 2002
2. Sieci komputerowe TCP/IP. 3, Programowanie w trybie klient-serwer, wersja BSD, Comer D., Stevens D., Wydawnictwa Naukowo-Techniczne, 1997
3. Computer Networks, A. S. Tanenbaum, Pearson, 2014

| | | |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|-------------|
| Literatura uzupełniająca: | | |
| 1. Data and Computer Communications. Networking and Internetworking, Hura G. S., Singhal M., CRC Press LLC, 2001 | | |
| 2. Client/Server Survival Guide, Harkey D., Wiley, 1999 | | |
| 3. Wireless Communications and Networks, Stallings W., Pearson, Prentice Hall, 2002 | | |
| 4. Algorithms and Protocols for Wireless, Mobile Ad Hoc Networks, Boukerche A., Wiley-IEEE Press, 2008 | | |
| Bilans nakładu pracy przeciętnego studenta | | |
| Czynność | Czas (godz.) | |
| 1. udział w zajęciach laboratoryjnych: | 30 | |
| 2. przygotowanie do zajęć laboratoryjnych: | 3 | |
| 3. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności ćwiczeń / projektu (częściowo mogą być realizowane drogą elektroniczną): | 2 | |
| 4. przygotowanie do sprawdzianów / kolokwium: | 2 | |
| 5. udział w wykładach: | 30 | |
| 6. zapoznanie się ze wskazaną literaturą / materiałami dydaktycznymi (10 stron tekstu naukowego = 1 godz.), 50 stron: | 5 | |
| 7. przygotowanie do zaliczenia: | 6 | |
| Obciążenie pracą studenta | | |
| forma aktywności | godzin | ECTS |
| Łączny nakład pracy | 78 | 4 |
| Zajęcia wymagające bezpośredniego kontaktu z nauczycielem | 60 | 3 |
| Zajęcia o charakterze praktycznym | 18 | 1 |