



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Systemy rozproszone dużej skali [S2Inf1-SRC>SRDS]

Przedmiot

Kierunek studiów
Informatyka

Rok/Semestr
1/2

Studia w zakresie (specjalność)
Systemy rozproszone i chmurowe

Profil studiów
ogólnoakademicki

Poziom studiów
drugiego stopnia

Język oferowanego przedmiotu
polski

Forma studiów
stacjonarne

Wymagalność
obligatoryjny

Liczba godzin

Wykład
30

Laboratorium
30

Inne
0

Ćwiczenia
0

Projekty/seminaria
0

Liczba punktów ECTS

5,00

Koordynatorzy

dr inż. Tadeusz Kobus
tadeusz.kobus@put.poznan.pl

dr hab. inż. Anna Kobusińska prof. PP
anna.kobusinska@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu kształcenia ze studiów I stopnia zdefiniowanych w Uchwale Senatu PP, weryfikowane w procesie rekrutacji na studia 2 stopnia, efekty te prezentowane są w serwisie internetowym wydziału <https://cat.put.poznan.pl/>. W szczególności student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu systemów operacyjnych, przetwarzania rozproszonego, rozproszonych systemów operacyjnych oraz sieci komputerowych. Ponadto, student powinien posiadać także umiejętność pozyskiwania informacji ze wskazanych źródeł, jak również rozumieć konieczność poszerzania swoich kompetencji i mieć gotowość do podjęcia współpracy w ramach zespołu.

Cel przedmiotu

Przekazanie studentom podstawowej wiedzy związanej z wyzwaniami konstrukcji systemów rozproszonych dużej skali, w zakresie prezentacji teoretycznych i praktycznych aspektów konstrukcji rozproszonych systemów dużej skali. Prezentacja problematyki przetwarzania dużych ilości danych (big data) w systemach rozproszonych dużej skali. Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z przetwarzaniem w systemach rozproszonych dużej skali.

Przedmiotowe efekty uczenia się

Wiedza:

1. ma zaawansowaną i pogłębianą wiedzę z zakresu szeroko rozumianych systemów informatycznych, podstaw teoretycznych ich budowania oraz metod, narzędzi i środowisk programistycznych w zakresie algorytmów i złożoności, architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, języków i paradygmatów programowania
2. ma uporządkowaną i podbudowaną teoretycznie wiedzę ogólną związaną z kluczowymi zagadnieniami z zakresu informatyki takimi jak: architektura i klasyfikacja systemów rozproszonych dużej skali i ich środowiska komunikacyjnego, protokołów epidemicznych i plotkujących, systemów przetwarzania dużej ilości danych
3. ma zaawansowaną wiedzę szczegółową dotyczącą wybranych zagadnień z zakresu informatyki
4. ma zaawansowaną i szczegółową wiedzę o procesach zachodzących w cyklu życia systemów informatycznych sprzętowych lub programowych

Umiejętności:

1. 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
2. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych
3. potrafi dokonać krytycznej analizy istniejących rozwiązań technicznych oraz zaproponować ich ulepszenia (usprawnienia)
4. potrafi ocenić przydatność metod i narzędzi służących do rozwiązania zadania inżynierskiego, polegającego na budowie lub ocenie systemu informatycznego lub jego składowych, w tym dostrzec ograniczenia tych metod i narzędzi;
5. potrafi - stosując m.in. koncepcyjnie nowe metody - rozwiązywać złożone zadania informatyczne, w tym zadania nietypowe oraz zadania zawierające komponent badawczy
6. potrafi zgodnie z zadaną specyfikacją, uwzględniającą aspekty pozatechniczne zaprojektować złożone urządzenie, system informatyczny lub proces oraz zrealizować ten projekt co najmniej w części używając właściwych metod, technik i narzędzi, w tym przystosowując do tego celu istniejące lub opracowując nowe narzędzia

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe
2. rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu informatyki w rozwiązywaniu problemów badawczych i praktycznych

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wiedza nabyta w ramach wykładu jest weryfikowana na egzaminie pisemnym o charakterze problemowym, składającym się z 4-5 zadań otwartych. Za każde zadanie można otrzymać 10 pkt. Zaliczenie przedmiotu uzyskiwane jest w wyniku otrzymania min. 50% maksymalnej ilości punktów możliwych do uzyskania.

Umiejętności nabyte w ramach zajęć laboratoryjnych weryfikowane są podstawie:

- oceny przygotowania studenta do poszczególnych sesji zajęć (sprawdzian "wejściowy") ,
- ocenianie ciągle, na każdych zajęciach (odpowiedzi ustne),
- ocenę wiedzy i umiejętności związanych z realizacją zadania projektowego

Możliwe jest uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za omówienia dodatkowych aspektów zagadnienia. Próg zaliczeniowy: 50% punktów.

Treści programowe

Program wykładu obejmuje następujące zagadnienia: przedstawienie wyzwań związanych z budową i przetwarzaniem w systemach rozproszonych dużej skali, modele CAP i PACELC, wprowadzenie do tematyki baz danych NoSQL, przetwarzanie dużych danych z wykorzystaniem platformy Apache Spark, systemy zarządzania zasobami, rozproszony system plików i systemy kolejujące, architektury systemów P2P, architektura systemów Big Data, przykłady wykorzystania narzędzi dużych danych w istniejących systemach, technologia blockchain

Tematyka zajęć

Program wykładu obejmuje następujące zagadnienia:

1. Przedstawienie wyzwań związanych z budową i przetwarzaniem w systemach rozproszonych dużej skali: klasyfikacje systemów, idea działania, zastosowania, przykłady implementacji, wprowadzenie definicji masywnych danych, źródła dużych danych, aspekty przetwarzania dużych danych
2. Modele CAP i PACELC
3. Wprowadzenie do tematyki baz danych NoSQL: klasyfikacja ze względu na modele danych (key value, column-oriented, document-oriented, column-oriented, graph-oriented); budowa i działanie systemów NoSQL (data partitioning, load balancing, replication, data versioning, membership management, failure handling) na przykładzie Amazon Dynamo; Google BigTable, HBase, Cassandra.
3. Przetwarzanie dużych danych z wykorzystaniem Resilient Distributed Datasets (RDD) i platformy Apache Spark
4. Systemy zarządzania zasobami Mesos i YARN (architektura, algorytmy lokowania zasobów)
5. Rozproszony system plików i systemy kolejujące
6. Architektury systemów P2P (ustrukturalizowane, nieustrukturalizowane, hybrydowe); organizacja węzłów, topologie, skalowalność, równoważenie obciążeń, Distributed Hash Table (DHT),
9. Protokół Bittorrent i jego rozszerzenia: PEX, Mullitracker, DHT oraz magnet links
10. Architektura systemów Big Data: pozyskiwanie dużych danych, składowanie dużych danych, przetwarzanie wsadowe/przetwarzanie strumieni danych, analiza dużych danych; algorytmy rozproszone w przetwarzaniu dużych danych; przykłady wykorzystania narzędzi dużych danych w istniejących systemach (m.in. na przykładzie Google, Facebook, Yahoo, LinkedIn, Cloudera, Microsoft)
13. Technologia blockchain - zasada działania, przeznaczenie, inteligentne kontrakty na przykładzie platformy Ethereum

Program laboratorium obejmuje zagadnienia związane z problematyką przetwarzania dużych ilości danych w systemie Casandra:

1. Instalacja, konfiguracja, interfejs programistyczny, typy danych, podstawowe operacje dostępne w danym systemie.
2. Praktyczne wykorzystanie systemu: modelowanie danych; sortowanie, filtrowanie, opcje tabel, indeksy, kolekcje, liczniki, lightweight transactions i inne zaawansowane funkcje, stronicowanie wyników. Wykorzystanie znaczników czasowych. Poziomy spójności. Awaryjne węzłów, partycje sieciowe i resynchronizowanie zegarów. Konflikty znaczników czasowych.

Metody dydaktyczne

1. Wykład: prezentacja multimedialna, ilustrowana przykładami podawanymi na tablicy.
2. Zajęcia laboratoryjne: prezentacja multimedialna ilustrowana przykładami podawanymi na tablicy oraz demonstracja, dyskusja, warsztaty, ćwiczenia praktyczne, praca w zespole.

Literatura

Podstawowa

1. Distributed Systems: Principles and Paradigms, A. S. Tanenbaum, M. van Steen, Prentice-Hall, Inc, 2007
2. Peer-to-peer systems and applications, R. Steinmetz, K. Wehrle, Springer, 2005
3. NoSQL distilled, P. Sadalage, M. Flower, Addison-Wesley, 2013

Uzupełniająca

1. Spark in Action, Bonać M., Zečević P., Manning, 2015
2. Large Scale Network-Centric Distributed Systems, H.Sarbazi-Azad, A.Y.Zomaya, Wiley-IEEE Computer Society Press, 2013
3. M. Jelasity, S. Voulgaris, R. Guerraoui, A.-M. Kermarrec, M. Van Steen: Gossip-based peer sampling. ACM Trans. Comput. Syst 25(3) 2007
4. Nitin Sawant, Himanshu Shah, Big data application architecture Q&A, Springer, 2013
5. J.Berman, Principles of Big Data: Preparing, Sharing, and Analyzing Complex Information, Morgan-

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	125	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	65	2,50