



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Architektura systemów komputerowych

Przedmiot

Kierunek studiów

Teleinformatyka

Rok/semestr

1/2

Studia w zakresie (specjalność)

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszy

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obowiązkowy

Liczba godzin

Wykład

30

Laboratoria

30

Inne (np. online)

Ćwiczenia

0

Projekty/seminaria

0/0

Liczba punktów ECTS

5

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

prof. dr hab. inż. Ryszard Stasiński, Instytut
Telekomunikacji Multimedialnej, 61 665 3839,
ryszard.stasinski@put.poznan.pl

Odpowiedzialny za przedmiot/wykładowca:

dr Sławomir Michalak, Instytut Telekomunikacji
Multimedialnej, 61 665 3824

slawomir.michalak@put.poznan.pl

dr Krzysztof Arnold, Instytut Telekomunikacji
Multimedialnej, 61 665 3868

krzysztof.arnold@put.poznan.pl

Wymagania wstępne

Umiejętność programowania w języku C na poziomie elementarnym, wstępna wiedza z techniki cyfrowej.

Cel przedmiotu

Zapoznanie się ze stykiem sprzęt-oprogramowanie typowego komputera (ISA - Instruction Set Architecture) w celu zrozumienia zjawisk obniżających wydajność obliczeniową komputera przy kodowaniu algorytmów.



Przedmiotowe efekty uczenia się

Wiedza

Zapoznanie się ze stosowanymi we współczesnych procesorach technikami optymalizacji sprzętu (potokowość, równoległość na poziomie instrukcji, spekulatywne wykonywanie kodu), oraz hierarchią pamięci, w tym współpracą wielu procesorów z pamięcią główną.

Umiejętności

Programowanie w assemblerze procesora 32-bitowego (ARM, lub podobny), zapoznanie się z elementarnymi sposobami unikania zagrożeń potoku i optymalizacji kodu przy równoległości na poziomie instrukcji.

Kompetencje społeczne

Znajomość zasad tworzenia kodu na poziomie assemblera pozwalających na jego wykorzystanie przez innych użytkowników.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykład: egzamin końcowy, 10 równo punktowanych pytań wymagających opisowej odpowiedzi, próg zaliczenia: 5.1 pkt. na 10 możliwych. Ćwiczenia laboratoryjne: przeprowadzone prawidłowo ćwiczenia, ocena na podstawie sprawozdań i sprawdzenia wiedzy w trakcie wykonywania ćwiczeń, dwa kolokwia.

Treści programowe

WYKŁADY

Rewolucja komputerowa, pomiar wydajności komputerów, prawo Amdahla, najnowsze tendencje.

Typy instrukcji i ich opis, powiązanie instrukcji z podzespołami komputera, formaty instrukcji, powiązanie z językami wysokiego poziomu, implementacja struktur danych.

Realizacja podstawowych operacji arytmetycznych (dodawanie/odejmowanie, mnożenie, dzielenie) w wersji stałoprzecinkowej i zmiennoprzecinkowej. Błędy obliczeń zmiennoprzecinkowych a standard IEEE 754. Wynik obliczeń a kolejność operacji.

Fazy wykonania instrukcji i ich powiązania z podzespołami komputera. Potokowość: idea, prosty procesor pracujący sekwencyjnie i potokowo. Optymalizacja potoku. Zagrożenia potoku i sposoby ich omijania. Wyjątki i przerwania, ich wpływ na potok. Równoległość na poziomie instrukcji (ILP - Instruction Level Parallelism), podstawowe rozwiązania, wpływ na potok.

Hierarchia pamięci, skąd się wzięła. Sposoby realizacji pamięci podręcznych. Problemy związane z nietrafieniem pamięci podręcznej. Pamięć wirtualna (w pamięci masowej): organizacja, współpraca z pamięciami wyższego rzędu. Ochrona pamięci - tryby dostępu. Koherencja i spójność pamięci w systemach wieloprocessorowych.

Układy wejścia wyjścia i pamięć masowa: zachowanie, niezawodność. Technologie pamięci masowych. RAID.

Problemy z rozpisywaniem algorytmów na wiele procesorów: część sekwencyjna algorytmu, komunikacja. Wielowątkowość. Procesory wektorowe. Procesory graficzne - GPU.

ĆWICZENIA LABORATORYJNE

Arytmetyczne operacje stałoprzecinkowe

Inne operacje na danych jednostki stałoprzecinkowej

Operacje zmiennoprzecinkowe, cz. 1

Operacje zmiennoprzecinkowe, cz. 2

Realizacja instrukcji sterujących w assemblerze



Tablice w asemblerze

Tryb SIMD

Przekazywanie parametrów wywołania do funkcji cz. 1

Przekazywanie parametrów wywołania do funkcji cz. 2

Obsługa przerwań i wyjątków

Omijanie zagrożeń potoku

Programowanie wielowątkowe.

Metody dydaktyczne

Wykład - prelekcja, ćwiczenia na zestawach laboratoryjnych.

Literatura

Podstawowa

D. Patterson, J. Hennessy: Computer Organization and design, wyd. 4 (lub nowsze), Elsevier 2009.

Uzupełniająca

1. W. Stallings: Organizacja i architektura systemu komputerowego. WNT, Warszawa 2000.

2. J. Hennessy, D. Patterson: Computer Architecture: A Quantitative Approach, wyd. 4 (lub nowsze), 2011

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	116	5.0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	3.0
Praca własna studenta (przygotowanie do zaliczenia, przygotowanie do laboratorium, studia literaturowe)	56	2.0