**Nazwa przedmiotu:**

Architektura komputerów

**Koordynator przedmiotu:**

Dariusz TURLEJ, Grzegorz MAZUR

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Przedmioty techniczne

**Kod przedmiotu:**

ARKO

**Semestr nominalny:**

3 / rok ak. 2018/2019

**Liczba punktów ECTS:**

5

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

45 - wykłady
30 - nauka własna
15 - ćwiczenia laboratoryjne
40 - przygotowanie do ćwiczeń laboratoryjnych i praca nad projektami w domu

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

45 - wykłady
15 - ćwiczenia laboratoryjne
w sumie 60 godz., co daje ok. 2,5 ECTS

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

15 - ćwiczenia laboratoryjne
40 - przygotowanie do ćwiczeń laboratoryjnych i praca nad projektami w domu
w sumie 55 godz. co daje ok. 2 ECTS

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład: | 45h |
| Ćwiczenia: | 0h |
| Laboratorium: | 15h |
| Projekt: | 0h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

Programowanie w języku C, podstawy techniki cyfrowej, arytmetyka binarna.

**Limit liczby studentów:**

120

**Cel przedmiotu:**

Poznanie budowy i działania procesora i komputera. Nauka programowania asemblerowego i hybrydowego. Zapoznanie z aspektami wydajnościowymi w działaniu sprzętu i oprogramowania. Zapoznanie z architekturą systemową komputera.

**Treści kształcenia:**

Taksonomie architektur komputerów. Maszyna von Neumanna, architektury Harvard, Princeton, Harvard-Princeton. Pojęcie hierarchii pamięci.
Typy i formaty danych. Interpretacja danych binarnych. Kody używane do reprezentacji znaków pisarskich, liczb całkowitych, stało- i zmiennopozycyjnych. Adresowanie pamięci. Dane w pamięci - konwencje Big i Little Endian, wyrównanie danych. Dane wektorowe.
Wymagania języków wysokiego poziomu wobec architektury procesora. Identyfikacja klas pamięci. Stos. Przekazywanie sterowania pomiędzy procedurami – konwencje wołania. Mapa pamięci programu. Ramka stosu w rejestrach i w pamięci. Model programowy procesora — zestawy rejestrów, tryby adresowania, model operacji warunkowych. Konstrukcja listy instrukcji - podejście RISC i CISC. Formaty instrukcji. Prezentacja modelu programowego RISC i CISC.
Wykonanie instrukcji przez procesor. Procesor z jednofazowym wykonaniem instrukcji. Procesor wielofazowy z jednostką sterującą. Prefetch i kolejka instrukcji. Potok. Synchronizacja potoku – hazardy i opóźnienia. Superpotok. Potokowa implementacja procesora CISC – potok CISC, potok z transkodowaniem. Architektury superskalarne. Synchronizacja wielopotokowej jednostki wykonawczej. Prezentacja współczesnych architektur wielopotokowych z transkodowaniem instrukcji.
Kieszenie — zasada działania, warianty organizacji. Problem spójności zawartości pamięci operacyjnej i kieszeni. Analiza wydajności.
Problemy wydajnościowe w architekturach superskalarnych i superpotokowych. Przewidywaie skoków. Sterowanie działaniem kieszeni. Mechanizmy przyspieszające odwołania do danych. Procesory wielowątkowe. Fuzja instrukcji.
Wymagania systemu operacyjnego. Systemowy model programowy procesora. Zasady ochrony zasobów. Zarządzanie pamięcią. Funkcje systemu zarządzania pamięcią. Metody zarządzania pamięcią — prosta relokacja, segmentacja, stronicowanie. Optymalizacja stronicowania. Stronicowanie w procesorach 64-bitowych. Zarządzanie działaniem hierarchii pamięci — zarządzanie pamięcią a kieszenie - sprzężenie kieszeni i jednostki zarządzania pamięcią.
Wyjątki – definicja i klasyfikacja. Zasady obsługi wyjątków. Specyficzne sytuacje wyjątkowe — podwójny błąd i restart. Implementacja systemu pamięci wirtualnej. Przechwycenia w maszynie wirtulanej.
Budowa komputera. Podsystem wejścia-wyjścia. Zasady współpracy z urządzeniami wejścia-wyjścia - aktywne oczekiwanie, przerwania, zastosowanie bezpośredniego dostępu do pamięci.
Struktury komputerów – pamięciocentryczna, architektury jedno-, dwu- i trójszynowe, architektury z połączeniami punkt-punkt. Przegląd struktur komputerów osobistych.
Zakres laboratorium
Zasady programowania w językach asemblerowych. Programowanie hybrydowe. Konwencje wołania procedur. Przekazywanie parametrów.
Zadania projektowe: procedury assemblerowe dołączone do programu w języku wysokiego poziomu; algorytmy operacji arytmetycznych, optymalizacja kodu.
Prosty program w środowisku symulatora procesora MIPS (2h).
Program przetwarzania znaków lub liczb - MIPS (4h).
Zasady programowania hybrydowego procesorów x86 - prosty program C plus asembler) (2h).
Procedura asemblerowa realizująca nietrywialne przetwarzanie (np. implementacja wybranych algorytmów obliczeniowych z arytmetyki komputerów) łączona z programem w języku C (4h).

**Metody oceny:**

kolokwia, projekty oprogramowania

**Egzamin:**

nie

**Literatura:**

1. Patterson, Hennessy: Computer Patterson, Hennessy: Computer Organization and Design, The Hardware/Software Interface, Elsevier
2. J. Biernat, Arytmetyka Komputerów, Warszawa, PWN
3. G. Mazur, Architektura Komputerów - preskrypt do wykładu.
4. MIPS32 Architecture for Programmers, MIPS Technologies (www.mips.com)
5. Intel® 64 and IA-32 Architectures Software Developer's Manual, Intel Corp.
6. AMD64 Architecture Programmer's Manual, AMD Inc.

**Witryna www przedmiotu:**

https://usosweb.usos.pw.edu.pl/kontroler.php?\_action=katalog2/przedmioty/pokazPrzedmiot&prz\_kod=103B-INxxx-ISP-ARKO

**Uwagi:**

## Charakterystyki przedmiotowe

### Profil ogólnoakademicki - wiedza

**Charakterystyka ARKO\_W01:**

Rozumie zasadę działania komputera i proces wykonywania programu przez sprzęt.

Weryfikacja:

Testy, zadania projektowe.

**Powiązane charakterystyki kierunkowe:** K\_W08, K\_W09, K\_W14, K\_W19

**Powiązane charakterystyki obszarowe:** I.P6S\_WG, III.P6S\_WG.o

**Charakterystyka ARKO\_W02:**

Zna sposoby reprezentacji danych w komputerze i potrafi posługiwać się typami danych.

Weryfikacja:

Testy, egzamin, zadania projektowe.

**Powiązane charakterystyki kierunkowe:** K\_W09, K\_W11, K\_W14, K\_W19

**Powiązane charakterystyki obszarowe:** I.P6S\_WG, III.P6S\_WG.o

**Charakterystyka ARKO\_W03:**

Potrafi zidentyfikować ograniczenia wydajności oprogramowania wynikające z budowy sprzętu.

Weryfikacja:

Testy, egzamin, zadania projektowe

**Powiązane charakterystyki kierunkowe:** K\_W11, K\_W14

**Powiązane charakterystyki obszarowe:** I.P6S\_WG, III.P6S\_WG.o

**Charakterystyka ARKO\_W04:**

Rozumie zasady zarządzania zasobami komputera, zna rolę systemu operacyjnego i mechanizmy sprzętowe komputera związane z zarządzaniem i ochroną zasobów.

Weryfikacja:

testy, egzamin

**Powiązane charakterystyki kierunkowe:** K\_W09, K\_W10, K\_W14

**Powiązane charakterystyki obszarowe:** I.P6S\_WG, III.P6S\_WG.o

**Charakterystyka ARKO\_W05:**

Rozumie zasady współpracy komputera z urządzeniami peryferyjnymi i zna typowe struktury komputerów.

Weryfikacja:

testy, egzamin

**Powiązane charakterystyki kierunkowe:** K\_W08, K\_W14

**Powiązane charakterystyki obszarowe:** I.P6S\_WG, III.P6S\_WG.o

### Profil ogólnoakademicki - umiejętności

**Charakterystyka ARKO\_U01:**

Potrafi tworzyć programy w językach asemblerowych i programy hybrydowe.

Weryfikacja:

Projekty laboratoryjne.

**Powiązane charakterystyki kierunkowe:** K\_U15, K\_U21, K\_U09, K\_U13, K\_U14

**Powiązane charakterystyki obszarowe:** III.P6S\_UW.4.o, I.P6S\_UW, I.P6S\_UU, III.P6S\_UW.3.o

**Charakterystyka ARKO\_U02:**

Potrafi w podstawowym zakresie zidentyfikować ograniczenia wydajności i wskazać możliwości optymalizacji oprogramowania.

Weryfikacja:

Projekty laboratoryjne

**Powiązane charakterystyki kierunkowe:** K\_U09, K\_U13, K\_U20, K\_U21

**Powiązane charakterystyki obszarowe:** I.P6S\_UU, I.P6S\_UW, III.P6S\_UW.4.o, III.P6S\_UW.3.o

**Charakterystyka ARKO\_U03:**

Rozumie potrzebę pogłębiania wiedzy i korzystania z literatury technicznej.

Weryfikacja:

Projekty laboratoryjne

**Powiązane charakterystyki kierunkowe:** K\_UK01

**Powiązane charakterystyki obszarowe:** I.P6S\_UU