**SYLABUS**

**dotyczy cyklu kształcenia 2022-2026**

Rok akademicki 2023/2024

1. Podstawowe informacje o przedmiocie

|  |  |
| --- | --- |
| Nazwa przedmiotu | architektura systemów komputerowych |
| Kod przedmiotu |  |
| Nazwa jednostki prowadzącej kierunek | Kolegium Nauk Przyrodniczych |
| Nazwa jednostki realizującej przedmiot | Kolegium Nauk Przyrodniczych |
| Kierunek studiów | informatyka |
| Poziom studiów | studia inżynierskie I-go stopnia |
| Profil | ogólnoakademicki |
| Forma studiów | stacjonarne |
| Rok i semestr studiów | rok II, semestr 4 |
| Rodzaj przedmiotu | inżynierski przedmiot kierunkowy |
| Język wykładowy | język polski, język angielski |
| Koordynator | dr Krzysztof Balicki |
| Imię i nazwisko osoby prowadzącej / osób prowadzących | dr Krzysztof Balicki, mgr inż. Jarosław Szkoła |

1.1.Formy zajęć dydaktycznych, wymiar godzin i punktów ECTS

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Semestr  (nr) | Wykł. | Ćw. | Konw. | Lab. | Sem. | ZP | Prakt. | Inne (jakie?) | **Liczba pkt ECTS** |
| 4 | 15 |  |  | 30 |  |  |  |  | 4 |

1.2. Sposób realizacji zajęć

zajęcia w formie tradycyjnej

1.3 Forma zaliczenia przedmiotu / modułu (z toku)

zaliczenie z oceną

1. Wymagania wstępne

|  |
| --- |
| Podstawy programowania w języku C. |

1. Cele, efekty uczenia się, treści programowe i stosowane metody dydaktyczne

3.1 Cele przedmiotu

|  |  |
| --- | --- |
| C1 | Celem przedmiotu jest zapoznanie słuchaczy z architekturą współczesnych systemów komputerowych ze szczególnym uwzględnieniem procesora i koprocesora arytmetycznego. |
| C2 | Ważną cechą tych zajęć jest duża liczba przykładów i ćwiczeń praktycznych, które umożliwiają studentom poznanie podstaw programowania niskopoziomowego. |

**3.2 Efekty uczenia się dla przedmiotu**

|  |  |  |
| --- | --- | --- |
| EK (efekt uczenia się) | Treść efektu uczenia się zdefiniowanego dla przedmiotu | Odniesienie do efektów kierunkowych |
| EK\_01 | Student posiada wiedzę na temat organizacji systemu komputerowego oraz architektury mikroprocesora  i innych podzespołów jednostki systemowej. | K\_W03 |
| EK\_02 | Student zna narzędzia i techniki wykorzystywane  w programowaniu niskopoziomowym takie jak asembler, linker, disasembler, debugger i asmloader. Potrafi łączyć kod w asemblerze z programami w języku C. Zna sposoby komunikacji procesora z koprocesorem arytmetycznym  i jednostkami wektorowymi. | K\_W07 |
| EK\_03 | Student posiada umiejętność programowania procesora  i koprocesora arytmetycznego w stopniu podstawowym. | K\_U05 |

**3.3 Treści programowe**

1. Problematyka wykładu

|  |
| --- |
| Treści merytoryczne |
| Wprowadzenie – system komputerowy, klasyfikacja architektur komputerowych, hierarchia pamięci. Maszyna von Neumanna, architektury Harvard, Princeton, Harvard-Princeton. |
| Dane – typy, reprezentacje, organizacja i adresowanie pamięci. Porządek Big-Endian i Little-Endian. Wyrównanie naturalne. Dane wektorowe. |
| Budowa modelu programowego — rejestry, tryby adresowania, model operacji warunkowych, lista instrukcji. Porównanie modelu programowego w podejściu CISC i RISC. |
| Model programowy procesorów 16, 32 i 64 bitowych rodziny Intela. |
| Arytmetyka stało i zmiennopozycyjna. Jednostki zmiennopozycyjne i wektorowe. |
| Wybrane konwencje wywołań dla kodu 32-bitowego. |
| Konwencje wywołań dla kodu 64-bitowego. |
| Analiza stanu stosu procesora i koprocesora arytmetycznego. |
| Analiza programów rekurencyjnych, nierekurencyjnych i niskopoziomowych. |
| Asembler NASM, asmloader, disasembler, debugger GDB, linker. |

1. Problematyka ćwiczeń audytoryjnych, konwersatoryjnych, laboratoryjnych, zajęć praktycznych

|  |
| --- |
| Treści merytoryczne |
| Konwersje liczb między różnymi systemami liczbowymi. |
| Reprezentacja danych, konwencje little-endian i big-endian. |
| Mapa pamięci procesu w języku C. |
| Obsługa programów: NASM, asmloader, linker, disasembler, debugger DBG. |
| Pseudoinstrukcje asemblera NASM. |
| Operacje przesłań. |
| Tryby adresowania. |
| Operacje arytmetyczne, logiczne, porównań, bitowe, przesunięcia i rotacje bitów. |
| Instrukcje skoków, skoków warunkowych, wywołania procedur. |
| Analiza stanu stosu procesora i koprocesora arytmetycznego. |
| Analiza programów rekurencyjnych, nierekurencyjnych i niskopoziomowych. |
| Łączenie kodu w asemblerze z programami w języku C. |
| Mierzenie czasu wykonania procedur i funkcji. |

3.4 Metody dydaktyczne

Wykład: wykład z prezentacją multimedialną.

Laboratorium: rozwiązywanie zadań, projekt.

1. Metody i kryteria oceny

4.1 Sposoby weryfikacji efektów uczenia się

|  |  |  |
| --- | --- | --- |
| Symbol efektu | Metody oceny efektów uczenia się  (np.: kolokwium, egzamin ustny, egzamin pisemny, projekt, sprawozdanie, obserwacja w trakcie zajęć) | Forma zajęć dydaktycznych  (w, ćw, …) |
| EK\_01 | obserwacja w trakcie zajęć, sprawdzian | lab, w |
| EK\_02 | obserwacja w trakcie zajęć, sprawdzian | lab, w |
| EK\_03 | kolokwium, projekt | lab |

4.2 Warunki zaliczenia przedmiotu (kryteria oceniania)

|  |
| --- |
| Wykład  Zaliczenie bez oceny. Sprawdzian na ostatnim wykładzie. Aby zaliczyć sprawdzian, należy zdobyć przynajmniej połowę maksymalnej liczby punktów.  Laboratorium  Warunkiem zaliczenia laboratorium jest zaliczenie kolokwium i wykonanie projektu. Ocena końcowa jest średnią ocen z kolokwium i projektu. Aby zaliczyć kolokwium, należy zdobyć przynajmniej połowę maksymalnej liczby punktów. Oceny z kolokwiów przyznawane są proporcjonalnie do liczby zdobytych punktów. Do tematów projektów przypisane są oceny referencyjne zależne od stopnia ich trudności. Ocena z obrony projektu może różnić się od oceny referencyjnej o pół stopnia. Pod uwagę brana jest również aktywność na zajęciach, która może obniżyć lub podwyższyć ocenę końcową o pół stopnia. |

1. Całkowity nakład pracy studenta potrzebny do osiągnięcia założonych efektów   
   w godzinach oraz punktach ects

|  |  |
| --- | --- |
| **Forma aktywności** | **Średnia liczba godzin na zrealizowanie aktywności** |
| Godziny kontaktowe wynikające z harmonogramu studiów | 45 |
| Inne z udziałem nauczyciela  (udział w konsultacjach, egzaminie) | 2 |
| Godziny niekontaktowe – praca własna studenta  (przygotowanie do zajęć, egzaminu, napisanie referatu itp.) | 53 |
| SUMA GODZIN | 100 |
| **SUMARYCZNA LICZBA PUNKTÓW ECTS** | 4 |

1. Praktyki zawodowe w ramach przedmiotu

|  |  |
| --- | --- |
| wymiar godzinowy | - |
| zasady i formy odbywania praktyk | - |

1. Literatura

|  |
| --- |
| Literatura podstawowa:   1. D. Patterson, J. Hennessy, Computer Organization and design, Elsevier 2005 2. J. Biernat, Architektura komputerów, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2005 |
| Literatura uzupełniająca:   1. P. Metzger, M. Siemieniecki, Anatomia PC : architektura komputerów zgodnych z IBM PC, Helion, Gliwice 2003 2. Materiały firmowe - dokumenty techniczne dostępne w sieci WWW – MIPS, Intel, AMD 3. Specyfikacje: Application Binary Interface |

Akceptacja Kierownika Jednostki lub osoby upoważnionej