**KONSPEKT PRZEDMIOTU**

Semestr letni, rok akad. 2020/2021

|  |  |
| --- | --- |
| Nazwa przedmiotu | Architektura systemów komputerowych |
| Kierunek/-i studiów / rok studiów / semestr studiów | Informatyka/1/II |
| Tryb studiów | Stacjonarny |
| Forma zajęć | Laboratorium |
| Liczba godzin | 30 |
| Koordynator przedmiotu: | dr hab. prof. AEH Mikołaj Aleksiejuk |
| Jakie są ogólne cele dydaktyczne tego przedmiotu? | Zdobycie wiedzy w zakresie: 1. podstaw działania i budowy układów cyfrowych, 2. zasady pracy systemu komputerowego, w tym podstawowych składowych: procesora, pamięci, podzespołów wejścia-wyjścia 3. architektury systemów komputerowych |
| Jak są kryteria zaliczenia tego przedmiotu? | Uzyskanie końcowej punktacji z laboratorium 20 pkt-ów przy spełnieniu następujących warunków:  1. zaliczyć 2 kolokwia uzyskać z każdego co najmniej 5 punktów.  2. zaliczyć 1 kolokwium z asemblera uzyskując co najmniej 7 pkt-ów  3. przy poprawkach obniża się uzyskaną punktacje o 20%  Z laboratorium wystawiana jest ocena zal/nzal. Uzyskana na laboratorium punkty stanowią część składową oceny końcowej z przedmiotu |
| Jakie są kryteria zaliczenia tego przedmiotu na ocenę celującą? | Zaliczenie przedmiotu na ocenę celującą wymaga przedstawienia projektu końcowego znacząco wykraczającego poza kryteria na ocenę bardzo dobrą oraz uzyskanie co najmniej 85% z max ilości możliwych do uzyskania pkt-ów. |
| Pozostałe informacje, dotyczące tego przedmiotu, ważne dla studenta | Do egzaminu dopuszczeni są studenci, którzy mają zaliczone laboratoria |

|  |  |
| --- | --- |
| **ZAJĘCIA 1**  **(3 godz.)**  **Systemy liczbowe** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:   1. Student będzie znał podstawowe systemy liczbowe 2. Student będzie potrafił dokonać konwersji liczb między różnymi systemami liczbowymi 3. Student będzie umiał wykonywać operacje arytmetyczne na liczbach w kodach binarnych. |
| Treści zajęć | 1. Podstawowe systemy liczbowe: binarne, o różnych podstawach 3,4, 8 oraz heksagonalny  2. Konwersje liczb dziesiętnych całkowitych i ułamkowych na kody binarne i heksagonalny  3. Operacje arytmetyczne na liczbach zapisanych w kodach binarnych |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Biernat J., Arytmetyka komputerów, 1996  Rozszerzające / uzupełniające:  - Stallings W., Organizacja i architektura systemu komputerowego 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 2**  **(3 godz.)**  **Systemy liczbowe. Format zmiennoprzecinkowy liczb wymiernych.** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1.Student będzie umiał przechodzić miedzy różnymi systemami liczb binarnych.  2. Student będzie znał kody binarne znaków (ASCI, Unicode)  3. Student będzie umiał kodować liczby wymierne w formacie zmiennoprzecinkowym IEEE754 |
| Treści zajęć | 1. Kodowanie liczb wymiernych w formacie zmiennoprzecinkowym IEEE754  2. Operacje arytmetyczne na liczbach w formacie zmiennoprzecinkowym  3. Kodowanie znaków |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Biernat J., Arytmetyka komputerów, 1996  Rozszerzające / uzupełniające:  - Stallings W., Organizacja i architektura systemu komputerowego 2009  - https://www.h-schmidt.net/FloatConverter/IEEE754.html |

|  |  |
| --- | --- |
| **ZAJĘCIA 3**  **(3 godz.)**  **Kod BCD. Maski bitowe.** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:   1. Student będzie znał kod BCD i potrafił dokonać konwersję liczb dziesiętnych na kod BCD 2. Student będzie umiał ustawiać pojedyncze bity w słowie bitowym. 3. Student będzie umiał sprawdzać ustawienie pojedynczych bitów w słowie bitowym. |
| Treści zajęć | 1. Konwersje liczb dziesiętnych całkowitych na kod BCD.  2. Maski bitowe |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Biernat J., Arytmetyka komputerów, 1996  Rozszerzające / uzupełniające:  - Barczak A., Florek J., Sydoruk T., Elektroniczne Techniki Cyfrowe, 2006  - Stallings W., Organizacja i architektura systemu komputerowego 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 4**  **(3 godz.)**  **Wyrażenia boolowskie. Cyfrowe układy kombinacyjne.** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie znał podstawowe prawa algebry Boole’a.  2. Student będzie umiał minimalizować funkcje boolowskie.  3. Student będzie umiał zaprojektować prosty układ kombinacyjny. |
| Treści zajęć | 1. Kolokwium z zadań realizowanych na zajęciach 1 - 3.  2. Metody opisu i minimalizacji funkcji boolowskich: tablica prawdy, formuła koniunkcyjno-alternatywna, formuła alternatywno-koniunkcyjnia,  3. Tworzenie prostych układów kombinacyjnych przy pomocy symulatora Multimedia Logic (lub Digital Works) i weryfikacja ich działania |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Barczak A., Florek J., Sydoruk T., Elektroniczne Techniki Cyfrowe, 2006  - Biernat J., Arytmetyka komputerów, 1996  Rozszerzające / uzupełniające:  - Stallings W., Organizacja i architektura systemu komputerowego, 2009  - Digital Works jako narzędzie wspomagające kształcenie inżynierów informatyki w zakresie symulacji cyfrowych układów logicznych, 2019, DOI: 10.15584/eti.2019.3.37 |

|  |  |
| --- | --- |
| **ZAJĘCIA 5**  **(3 godz.)**  **Układy kombinacyjne (cd). Sekwencyjne układy cyfrowe.** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie potrafił zbudować średnio złożone układy kombinacyjne  2. Student będzie rozumiał podstawowe idee sterowania sprzętowego w systemie komputerowym.  3. Student będzie znał budowę podstawowych układów sekwencyjnych |
| Treści zajęć | 1. Mapy Karnaugh’a  2. Sekwencyjne układy cyfrowe  3. Abstrakcyjny model sekwencyjnych układów cyfrowych  3. Zasadnicze typy układów sekwencyjnych: przerzutniki, rejestry, liczniki  4. Podanie zadanie dodatkowe dla studentów starających się o ocenę celującą |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Barczak A., Florek J., Sydoruk T., Elektroniczne Techniki Cyfrowe, 2006  - Biernat J., Arytmetyka komputerów, 1996  Rozszerzające / uzupełniające:  - Stallings W., Organizacja i architektura systemu komputerowego, 2009  - Digital Works jako narzędzie wspomagające kształcenie inżynierów informatyki w zakresie symulacji cyfrowych układów logicznych, 2019, DOI: 10.15584/eti.2019.3.37 |

|  |  |
| --- | --- |
| **ZAJĘCIA 6**  **(3 godz.)**  **Wprowadzenie do języka asembler** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie umiał posługiwać się środowiskiem MASM32  2. Student będzie umiał napisać prosty program w języku asembler |
| Treści zajęć | 1. Kolokwium z zadań realizowanych na zajęciach 4 - 5.  2. Wprowadzenie do środowiska masm32, kompilacja, łączenie  3. Uruchomianie programu w masm32 |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Irvine K. R.. Asembler dla procesorów Intel, 2003  - manual do masm32  Rozszerzające / uzupełniające:  - Stallings W., Organizacja i architektura systemu komputerowego 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 7**  **(3 godz.)**  **Asembler. Analiza stanu i ustawianie flag procesora** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie umiał ustawiać flagi procesora dostępne w rejestrze stanu procesora.  2. Student będzie znał szczegóły działania niektórych rozkazów procesora: arytmetycznych, logicznych, przesunięć i rotacji. |
| Treści zajęć | 1. Ustawianie flag procesora w języku asembler.  2. Samodzielna realizacja zadania programistycznego z zakresu ustawiania flag procesora.. |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Irvine K. R.. Asembler dla procesorów Intel. 2003  - manual do masm32  Rozszerzające / uzupełniające:  - Stallings W. Organizacja i architektura systemu komputerowego 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 8**  **(3 godz.)**  **Wykonywanie programów. Skoki. Makra. Procedury** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie rozumiał na czym polega rozkazy skoku i rozgałęzienia  2. Student będzie umiał stosować skoki, rozgałęzienia i instrukcje warunkowe w swoim programie w języku asembler  3. Student będzie umiał napisać w asemblerze prosty program z wykorzystaniem skoku, procedury i makr  4. Student będzie znał rodzaje przerwań |
| Treści zajęć | 1. Skoki i rozgałęzienia.  2. Realizacja zadania programistycznego ze skokami i instrukcjami warunkowymi  3. Procedury i makra  4. Przerwania |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Irvine K. R., Asembler dla procesorów Intel, 2003  - manual do masm32  Rozszerzające / uzupełniające:  - Stallings W.. Organizacja i architektura systemu komputerowego 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 9**  **(3 godz.)**  **Pobieranie danych z pamięci i wysyłanie danych do pamięci** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie znał podstawowe rodzaje pamięci  2. Student będzie znał w jaki sposób odbywa się pobieranie danych z pamięci  3. Student będzie znał w jaki sposób odbywa się wysyłanie danych do pamięci |
| Treści zajęć | 1. Pamięć podręczna  2. Realizacja zadania programistycznego pobrania i wysłania danych do pamięci  3. Podanie zadanie dodatkowe dla studentów starających się o ocenę celującą |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Irvine K. R., Asembler dla procesorów Intel, 2003  - Manual do masm32  Rozszerzające / uzupełniające:  - Stallings W.. Organizacja i architektura systemu komputerowego 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 10**  **(3 godz.)**  **Repetytorium. Weryfikacja zadań dodatkowych.** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie miał możliwość uzupełnienia swoich umiejętności z zakresu arytmetyki komputerów, projektowania układów kombinacyjnych oraz programowania w języku asembler. |
| Treści zajęć | 1. Sprawdzian poprawkowy z zakresu kolokwium 1 i 2 oraz asemblera. 2. Weryfikacja zadań domowych dodatkowych. |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Biernat J., Arytmetyka komputerów, 1996  - Barczak A., Florek J., Sydoruk T., Elektroniczne Techniki Cyfrowe, 2006  - Irvine K. R., Asembler dla procesorów Intel, 2003  Rozszerzające / uzupełniające:  - Stallings W., Organizacja i architektura systemu komputerowego 2009 |