**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 | Niestacjonarny |
| Forma zajęć | Laboratorium |
| Liczba godzin | 16 |
| Koordynator przedmiotu: | mgr inż. Łukasz Laszko |
| 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? | 1. Zaliczyć wszystkie zadania ze sprawdzianów na minimum 4 pkt.,  2. Zaliczyć zadanie projektowe na minimum 4 pkt.,  3. Zaliczyć minimum 1 zadanie z asemblera na minimum 4 pkt.  4. Zadania zaliczone w terminie podstawowym są warte maks. 10 pkt.  5. Zadania zaliczone w terminie poprawkowym są warte maks. 8 pkt.  Za laboratoria wystawiana jest ocena uogólniona zal/nzal. Zebrane punkty stanowią część składową oceny końcowej z przedmiotu. |
| Jakie są kryteria zaliczenia tego przedmiotu na ocenę celującą? | Uzyskanie z laboratoriów co najmniej 85% z 90 możliwych punktów oraz wykonanie w pełni dodatkowego zadania projektowego, a także uzyskanie na egzaminie co najmniej 85% z 50 możliwych punktów. |
| Pozostałe informacje, dotyczące tego przedmiotu, ważne dla studenta |  |

|  |  |
| --- | --- |
| **ZAJĘCIA 1**  **(2 godz.)**  **Systemy liczbowe** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie umiał przechodzić miedzy różnymi kodami systemu binarnego.  2. Student będzie umiał wykonywać operacje arytmetyczne na liczbach w kodach binarnych. |
| Treści zajęć | 1. Konwersje liczb dziesiętnych całkowitych i ułamkowych na kody binarne.  2. Operacje arytmetyczne na liczbach binarnych. Kod NKB i U2. |
| 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**  **(2 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 kodami systemu binarnego.  2. Student będzie umiał wykonywać operacje arytmetyczne na liczbach w kodach binarnych.  3. Student będzie umiał kodować liczby wymierne w formacie zmiennoprzecinkowym IEEE754 |
| Treści zajęć | 1. Konwersje liczb dziesiętnych całkowitych na kod BCD.  2. Operacje arytmetyczne na liczbach binarnych. Kod BCD.  3. Kodowanie liczb wymiernych w formacie zmiennoprzecinkowym IEEE754 |
| 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**  **(2 godz.)**  **Sprawdzian 1.  Maski bitowe** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:   1. Student będzie umiał ustawiać pojedyncze bity w słowie bitowym. 2. Student będzie umiał sprawdzać ustawienie pojedynczych bitów w słowie bitowym. 3. Student będzie rozumiał znaczenie masek bitowych w programowaniu komputerów. |
| Treści zajęć | 1. Sprawdzian z zadań realizowanych na zajęciach 1 i 2.  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**  **(2 godz.)**  **S****ynteza układów kombinacyjnych** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie umiał zaprojektować prosty układ kombinacyjny.  2. Student będzie umiał minimalizować funkcje boolowskie.  3. Student będzie rozumiał podstawowe idee sterowania sprzętowego w systemie komputerowym. |
| Treści zajęć | 1. Synteza układów kombinacyjnych.  2. Metody opisu i minimalizacji funkcji boolowskich: tablica prawdy, formuła koniunkcyjno-alternatywna, formuła alternatywno-koniunkcyjnia, tablica Karnaugh’a.  3. Wydanie i omówienie zadania projektowego z zakresu syntezy układów kombinacyjnych.  4. Wydanie i omówienie zadania dodatkowego dla osób starających się o ocenę celującą z przedmiotu. |
| 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**  **(2 godz.)**  **Sprawdzian 2. Wprowadzenie do języka asembler dla procesora x86** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie umiał napisać prosty program w języku asembler dla procesora x86.  2. Student będzie umiał posługiwać się środowiskiem TASM.  3. Student będzie umiał analizować program w czasie jego działania. |
| Treści zajęć | 1. Sprawdzian z zadań realizowanych na zajęciach 3 i 4.  2. Wprowadzenie do środowiska TASM, kompilacja, łączenie.  3. Analiza programu w środowisku Turbo Debugger. |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Irvine K. R.. Asembler dla procesorów Intel, 2003  - Syck G., Turbo Assembler. Biblia użytkownika, 2002  Rozszerzające / uzupełniające:  - Stallings W. Organizacja i architektura systemu komputerowego, 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 6**  **(2 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 analizować stan procesora w czasie wykonania programu.  2. Student będzie umiał ustawiać flagi procesora dostępne w rejestrze stanu procesora.  3. Student będzie znał szczegóły działania niektórych rozkazów procesora: arytmetycznych, logicznych, przesunięć i rotacji. |
| Treści zajęć | 1. Analiza stanu procesora po wykonaniu niektórych rozkazów.  2. Ustawianie flag procesora w języku asembler.  3. 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  - Syck G., Turbo Assembler. Biblia użytkownika, 2002  Rozszerzające / uzupełniające:  - Biernat J., Arytmetyka komputerów, 1996  - Stallings W. Organizacja i architektura systemu komputerowego, 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 7**  **(2 godz.)**  **Asembler. Korzystanie z funkcji systemowych** | |
| Po tych zajęciach student (będzie znał/ wiedział/ umiał/ potrafił / rozumiał) | Efekty uczenia się:  1. Student będzie umiał wykorzystać tablice w swoim programie w języku asembler.  2. Student będzie umiał korzystać z funkcji systemowych (INT 21h).  3. Student będzie umiał stosować skoki, rozgałęzienia i instrukcje warunkowe w swoim programie w języku asembler. |
| Treści zajęć | 1. Segment danych: zmienne proste i tablicowe.  2. Obsługa wejścia i wyjścia w programie. Funkcja 1h, 2h, 9h i 0Ah przerwania 21H.  3. Sterowanie wykonywaniem się programu: skoki, rozgałęzienia i instrukcje warunkowe.  4. Samodzielna realizacja zadania programistycznego z zakresu korzystania z funkcji systemowych. |
| Źródło/a do nauki treści z tych zajęć | Minimalne / obowiązkowe:  - Irvine K. R., Asembler dla procesorów Intel, 2003  - Syck G., Turbo Assembler. Biblia użytkownika, 2002  Rozszerzające / uzupełniające:  - Biernat J., Arytmetyka komputerów, 1996  - Stallings W. Organizacja i architektura systemu komputerowego, 2009 |

|  |  |
| --- | --- |
| **ZAJĘCIA 8**  **(2 godz.)**  **Repetytorium. Obrona wydanych zadań.** | |
| 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 sprawdzianów 1 i 2.  2. Obrona wydanych zadań domowych. |
| 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  - Syck G., Turbo Assembler. Biblia użytkownika, 2002  Rozszerzające / uzupełniające:  - Stallings W., Organizacja i architektura systemu komputerowego 2009 |