



**Akademia Nauk Stosowanych**  
**im. Hipolita Cegielskiego w Gnieźnie Uczelnia Państwowa**

**SYLABUS**

|                                  |  |   |
|----------------------------------|--|---|
| Pozycja przedmiotu w planie:     |  | R.I/S.II - 7  |
| <b>1. OGÓLNY OPIS PRZEDMIOTU</b> |  |   |
| 1                                | Nazwa modułu   | Moduł treści kierunkowych   |
| 2                                | Nazwa przedmiotu   | Architektura komputerów   |
| 3                                | Kierunek studiów   | Informatyka   |
| 4                                | Poziom studiów   | pierwszy  |
| 5                                | Forma studiów  | niestacjonarne  |
| 6                                | Profil studiów   | praktyczny  |
| 7                                | Rok studiów  | pierwszy  |
| 8                                | Semestr przedmiotu   | drugi   |
| 9                                | Jednostka prowadząca kierunek studiów  | Instytut Nauk Technicznych  |
| 10                               | Liczba punktów ECTS  | 3   |
| 11                               | Sposób zaliczenia:   | <b>wykład:</b> egzamin<br><b>laboratorium:</b> zaliczenie z oceną   |
| 12                               | Imię i nazwisko nauczyciela (li) akademickiego (ich), stopień lub tytuł naukowy, adres e-mail                                | mgr inż. Mateusz Leszek<br><i>m.leszek@ans-gniezno.edu.pl</i>   |
| 13                               | Imię i nazwisko koordynatora(ów) przedmiotu, stopień lub tytuł naukowy, adres e-mail   | mgr inż. Mateusz Leszek<br><i>m.leszek@ans-gniezno.edu.pl</i>   |
| 14                               | Język wykładowy  | polski  |
| 15                               | Tryb prowadzenia zajęć   | stacjonarny   |
| 16                               | Sposób prowadzenia zajęć   | -   |
| 17                               | Narzędzia informatyczne wykorzystywane do prowadzenia zajęć, udostępniania materiałów i komunikacji ze studentami            | Microsoft Teams / platforma Moodle  |
| 15                               | Przedmioty wprowadzające   | - Teoretyczne podstawy informatyki,<br>- Wprowadzenie do programowania,   |
| 16                               | Wymagania wstępne  | 1. Podstawowa wiedza o budowie i zasadzie działania systemu komputerowego.<br>2. Podstawowa znajomość składni i struktur dowolnego języka programowania.<br>3. Podstawowa wiedza o budowaniu prostych algorytmów. |
| 17                               | <b>Cele przedmiotu:</b>  |   |
| C1                               | Przekazanie studentom podstawowej wiedzy dotyczącej budowy i zasady działania systemów komputerowych.                        |   |
| C2                               | Przedstawienie podstawowych zasad programowania niskopoziomowego w języku assembler dla architektury IA-32.                  |   |
| C3                               | Nauka implementacji prostych problemów programistycznych z wykorzystaniem języka assembler i kompilatora NASM.               |   |
| 18                               | Forma zajęć, liczba godzin wymagająca bezpośredniego udziału nauczyciela akademickiego, liczba godzin nakładu pracy studenta |   |

| Forma zajęć   |  | Liczba godzin   |
|---|--|---|
| 1. wykład   |  | 16  |
| 2. laboratorium   |  | 8   |
| <b>Suma godzin</b>  |  | <b>24</b>   |
| L.p.  | Całkowity nakład pracy studenta  |   |
| 1.  | Nakład pracy związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela akademickiego wynosi:  | <b>Godzinowe obciążenie studenta</b>  |
|   | <b>wykład: 16 godz.</b>  | <b>26 godzin</b>  |
|   | <b>laboratorium: 8 godz.</b>   |   |
|   | <b>udział w egzaminie: 2 godz.</b>   |   |
| Nakład pracy związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela akademickiego wynosi <b>26 godzin</b> , co odpowiada <b>0,75 punktom ECTS</b> . |  |   |
| 2   | <p>Bilans nakładu pracy studenta:</p> <ul style="list-style-type: none"> <li>• przygotowanie do laboratoriów: 8 godzin,</li> <li>• samodzielne studiowanie tematów wykładowej: 15 godzin,</li> <li>• wykonywanie zadań domowych: 15 godzin,</li> <li>• przygotowanie do egzaminu: 10 godzin,</li> <li>• przygotowanie do zaliczenia z laboratoriów: 8 godzin,</li> </ul> <p>Łączny nakład pracy studenta wynosi <b>56 godzin</b>, co odpowiada <b>2,25 punktom ECTS</b>.</p> | <b>56 godzin</b>  |
| 3   | <b>Łączny nakład pracy studenta (pozycja 1+2)</b>  | 82 godzin   |
| 4   | <b>Punkty ECTS za przedmiot</b>  | <b>3 ECTS</b>   |
| 5   | Liczba punktów ECTS, którą student musi osiągnąć w ramach zajęć o charakterze praktycznym w tym zajęć laboratoryjnych, warsztatowych, projektowych   | 1 ECTS  |
| Efekty uczenia się - wiedza   |  | <b>K_W10:</b> student ma wiedzę z zakresu działania systemu komputerowego, arytmetyki maszyn cyfrowych, organizacji komputera na poziomie przestań międzyrejestrowych i poziomie instrukcji maszynowych; zna organizację i architektury systemów pamięci oraz działanie urządzeń wejścia/wyjścia i sposobu ich komunikacji w systemie komputerowym. |
|   |  | <b>K_W13:</b> student ma poszerzoną wiedzę w zakresie logicznej i fizycznej budowy komputera i urządzeń otoczenia procesora oraz pogłębione podstawy teoretyczne działania jego podstawowych elementów składowych (pamięć, CPU).  |
|   |  | <b>K_W18:</b> ma wiedzę o cyklu życia sprzętu komputerowego, stanie oraz najnowszych trendach i uwarunkowaniach rozwojowych informatyki w odniesieniu do architektury komputerów.   |
| Efekty uczenia się - umiejętności   |  | <b>K_U01:</b> student potrafi samodzielnie pozyskiwać informacje z literatury, baz danych i dokumentacji technicznych oraz efektywnie pozyskiwać wiedzę, w tym w systemie kształcenia zdalnego (blended/e-learning) w zakresie architektury komputerów  |
|   |  | <b>K_U09:</b> student ma umiejętność pisania prostych programów na poziomie asemblera z użyciem instrukcji warunkowych, pętli, operacji na liczbach całkowitych i ciągach znaków, tablic oraz stosu.  |
|   |  | <b>K_U13:</b> student potrafi pisać, uruchamiać, śledzić i testować programy środowisku programistycznym dla języka assembler notacji (NASM) wykorzystując znajomość paradygmatów i metod programowania niskopoziomowego dla architektury IA-32.  |
| Efekty uczenia się – kompetencje społeczne  |  | <b>K_K01:</b> student rozumie potrzebę i zna możliwości ciągłego dokształcania się, krytycznie odnosi się do posiadanej wiedzy, podnoszenia kompetencji zawodowych, osobistych i społecznych  |
|   |  | <b>K_K02:</b> student ma świadomość ważności i rozumie pozatechniczne aspekty i skutki działalności inżyniera-informatyka, w tym jej wpływ na środowisko i związaną z tym odpowiedzialność za podejmowane decyzje; dba o dobre tradycje zawodu informatyka.   |

| 2. TREŚCI PROGRAMOWE ODNIESIONE DO EFEKTÓW UCZENIA SIĘ |  |               |
|--|--|---------------|
|  | Treści programowe  | liczba godzin |
| <b>Forma: wykład</b>                                   |  |               |
| <b>W1</b>  | Podstawowe pojęcia: informacja cyfrowa, Model von Neumana, ALU   | 1,5           |
| <b>W2</b>  | Pamięci półprzewodnikowe   | 1,5           |
| <b>W3</b>  | Procesor - część I - ALU, CPU, Cykl rozkazowy, RISC i CISC   | 1,5           |
| <b>W4</b>  | Procesor - część II - Stos, Pamięć CACHE, Przetwarzanie potokowe i równoległe, HT i procesory wielordzeniowe       | 1,5           |
| <b>W5</b>  | Procesor - część III - Stronicowanie pamięci, tryby pracy procesora, rozszerzenia multimedialne, rozwój procesorów | 1,5           |
| <b>W6</b>  | Układy we/wy - jak je widzi procesor? System przerwań, DMA   | 1,5           |
| <b>W7</b>  | Układy otoczenia procesora - część I - Chipset; CPU vs. APU  | 1,5           |
| <b>W8</b>  | Układy otoczenia procesora - część II - Magistrale kart rozszerzeń   | 1,5           |
| <b>W9</b>  | Transmisja asynchroniczna; RS-232; USB   | 1,5           |
| <b>W10</b>   | Pamięci masowe; Modemy   | 1,5           |
| <b>Forma: laboratoria</b>                              |  |               |
| <b>L1</b>  | Wprowadzenie do asemblera i środowiska NASM  | 3             |
| <b>L2</b>  | Operatory logiczne i struktury sterujące   | 3             |
| <b>L3</b>  | Tablice i stos   | 3             |
| <b>L4</b>  | Ciągi znaków   | 3             |
| <b>L5</b>  | Operacje konwersji danych  | 1,5           |
| <b>L6</b>  | Funkcje zewnętrzne   | 1,5           |

| 3. Literatura                   |   |
|---------------------------------|---|
| <b>Literatura podstawowa</b>    | <ol style="list-style-type: none"> <li>1. Metzger P., Anatomia PC, wyd. IX, Helion, 2004 - <a href="https://biblioteka.ans-gniezno.edu.pl/?det&amp;oid=107&amp;dt=0">https://biblioteka.ans-gniezno.edu.pl/?det&amp;oid=107&amp;dt=0</a></li> <li>2. Dokumentacja architektury IA-32: <a href="https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html">https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html</a></li> <li>3. Dokumentacja kompilatora NASM: <a href="https://www.nasm.us/doc/">https://www.nasm.us/doc/</a></li> </ol> |
| <b>Literatura uzupełniająca</b> | <ol style="list-style-type: none"> <li>1. Stallings W., SYSTEMY OPERACYJNE Architektura, funkcjonowanie i projektowanie, wyd. IX, Helion, 2018 - <a href="https://biblioteka.ans-gniezno.edu.pl/?det&amp;oid=9613&amp;dt=0">https://biblioteka.ans-gniezno.edu.pl/?det&amp;oid=9613&amp;dt=0</a></li> <li>2. Kruk S., Asembler : wykłady i ćwiczenia, Mikom, 2003 - <a href="https://biblioteka.ans-gniezno.edu.pl/?det&amp;oid=537&amp;dt=1&amp;hdg=-1">https://biblioteka.ans-gniezno.edu.pl/?det&amp;oid=537&amp;dt=1&amp;hdg=-1</a></li> </ol>  |

| 4. Metody dydaktyczne |   |
|-----------------------|---|
| <b>Forma</b>          | <b>Metody dydaktyczne</b>   |
| wykład                | wykład informacyjny – prezentacja multimedialna, opowiadanie.                       |
| laboratoria           | Metody ćwiczeniowo - praktyczne, metoda doświadczeniowa – prezentacja multimedialna |

## 5. Metody i kryteria oceniania

|  |   |
|--|---|
| <b>Forma zajęć: wykład</b>   | <b>Forma zaliczenia: egzamin</b>            |
| Uzyskane punkty są przeliczane na oceny według następującej skali:<br><b>Procent punktów:</b> <b>Ocena:</b><br>91-100%                                      Bardzo dobry<br>85-90%                                        Dobry plus<br>76-84%                                        Dobry<br>66-75%                                        Dostateczny plus<br>51-65%                                        Dostateczny<br>0-50%                                         Niedostateczny |   |
| <b>Opis:</b> Egzamin w formie testu jednokrotnego wyboru na platformie Moodle.   |   |
| <b>Forma zajęć: laboratorium</b>   | <b>Forma zaliczenia: zaliczenie z oceną</b> |
| Uzyskane punkty są przeliczane na oceny według następującej skali:<br><b>Procent punktów:</b> <b>Ocena:</b><br>91-100%                                      Bardzo dobry<br>85-90%                                        Dobry plus<br>76-84%                                        Dobry<br>66-75%                                        Dostateczny plus<br>51-65%                                        Dostateczny<br>0-50%                                         Niedostateczny |   |
| <b>Opis:</b> Zaliczenie w formie pisemnej, rozwiązanie przedstawionego algorytmu w języku assembler (NASM– IA32).  |   |
| Warunkiem zaliczenia przedmiotu jest uzyskanie oceny pozytywnej ze wszystkich form zajęć.  |   |

| <b>Zatwierdzenie karty opisu zajęć</b> |   |               |
|--|---|---------------|
|  | <b>Stanowisko</b><br>Tytuł/stopień naukowy, imię nazwisko | <b>Podpis</b> |
| <b>Opracował</b>                       | mgr inż. Mateusz Leszek                                   |               |
| <b>Zatwierdził</b>                     | Dyrektor Instytutu Nauk Technicznych                      |               |