

Karta opisu zajęć - Sylabus

Państwowa Wyższa Szkoła Techniczno-Ekonomiczna im. ks. Bronisława Markiewicza w Jarosławiu

I. INFORMACJE PODSTAWOWE

Nazwa zajęć: Architektura systemów komputerowych	Cykl kształcenia: 2022/2023	Data aktualizacji sylabusa: 28.02.2022
Nazwa kierunku studiów, poziom i profil kształcenia: Informatyka I stopień, profil praktyczny		
Język wykładowy: polski	Rodzaj zajęć: przedmiot kształcenia kierunkowego	
Rok studiów: I	Semestr: II	
Liczba punktów ECTS przypisana zajęciom: 3	Koordynator zajęć Imię, nazwisko, tytuł/stopień naukowy, adres e-mail: Jan Cisek, dr inż., jan.cisek@pwste.edu.pl	
Jednostka organizacyjna: Instytut Inżynierii Technicznej	Prowadzący zajęcia Imię, nazwisko, tytuł/stopień naukowy, adres e-mail: Jan Cisek, dr inż., jan.cisek@pwste.edu.pl	

FORMA PROWADZENIA ZAJĘĆ I LICZBA GODZIN

Ogólna liczba godzin zajęć dydaktycznych na studiach stacjonarnych i niestacjonarnych z podziałem na formy:

Studia stacjonarne		Studia niestacjonarne	
Wykład:	30	Wykład:	
Ćwiczenia:		Ćwiczenia:	
Laboratorium:	30	Laboratorium:	
Lektorat:		Lektorat:	
Projekt:		Projekt:	
Zajęcia praktyczne:		Zajęcia praktyczne:	
Seminarium:		Seminarium:	
Zajęcia terenowe:		Zajęcia terenowe:	
Praktyki:		Praktyki:	
Inna forma (jaka):		Inna forma (jaka):	
RAZEM:	60	RAZEM:	

II. INFORMACJE SZCZEGÓŁOWE

Wymagania wstępne i dodatkowe:

- oznaczenia i sposób działania bramek logicznych i przerzutników,
- podstawy elektroniki i elektrotechniki
- umiejętność konstruowania algorytmów,
- deklaracje prostych (w tym wskaźników) i strukturalnych (tablic, unii) typów danych,
- umiejętność deklarowania funkcji.

Cel (cele) kształcenia dla zajęć:				
Celem zajęć jest przekazanie wiedzy na temat architektury systemów komputerowych na przykładzie wywodzących się z tradycyjnych konstrukcji komputerów klasy PC z procesorem 8086 i systemem operacyjnym DOS. Słuchacze są zapoznawani z takimi zagadnieniami jak: budowa wewnętrzna i działanie mikroprocesora 8086, budowa systemu mikroprocesorowego, system przerwań programowych (BIOS) i sprzętowych, system operacyjny DOS. Studenci praktycznie poznają zasady, techniki i narzędzia stosowane w programowaniu niskopoziomowym (assembler) uzyskując podstawową wiedzę w tym zakresie. Dodatkowo omawiane są współczesne konstrukcje komputerów klasy PC i tendencje rozwojowe w tym zakresie				
Efekty uczenia się określone dla zajęć				
Efekty uczenia się określone dla zajęć w kategorii wiedza, umiejętności oraz kompetencje społeczne oraz metody weryfikacji efektów uczenia się				
Symbol efektów uczenia się określonego dla zajęć		Po zakończeniu zajęć i potwierdzeniu osiągnięcia efektów uczenia się, student w kategorii:		
Wiedzy - zna i rozumie				
M_01	Zna budowę wewnętrzną mikroprocesora Intel8086 i architekturę sprzętową komputera. Rozumie podstawy organizacji systemu BIOS i DOS. Wie jak działają i współpracują ze sobą podstawowe komponenty komputera, w tym interfejsów wejścia-wyjścia. Rozumie znaczenie standaryzacji systemu operacyjnego w kontekście sprzętu komputerowego.			
M_02	Rozumie podstawy cyklu życia i trendy rozwojowe komputerów w aspekcie sprzętowym oraz programowym, także w zastosowaniach przemysłowych.			
Umiejętności - potrafi				
M_03	Potrafi indywidualnie realizować proste aplikacje. Umie oszacować czas potrzebny na realizację konkretnego zadania, potrafi opracować i zrealizować harmonogram prac.			
M_04	Potrafi użyć wybranego narzędzia programistycznego do pisania oraz testowania kodu prostej aplikacji.			
M_05	Umie analizować algorytmy pod względem ich poprawności i złożoności, a także potrafi skonstruować algorytm zgodny ze specyfikacją i zapisać go w wybranym języku programowania.			
Kompetencji społecznych - jest gotów do				
TREŚCI PROGRAMOWE I ICH ODNIESIENIE DO EFEKTÓW UCZENIA SIĘ OKREŚLONYCH DLA ZAJĘĆ				
Treści programowe (uszczegółowione, zaprezentowane z podziałem na poszczególne formy zajęć, tj. wykład, ćwiczenia, laboratoria, projekty, seminaria i inne):				
Symbol treści programowych	Opis treści programowych	Forma zajęć	Liczba godzin	Odniesienie do efektów uczenia się określonych dla zajęć (symbol efektów uczenia się)
		wykład		

TP_01	Podanie planu zajęć, zakresu przekazywanych wiadomości, wskazanie źródeł literaturowych i warunków uzyskania zaliczenia. Przypomnienie modelu maszyny RAM i sposobu jej programowania - elementarne instrukcje pseudokodu.	wykład podający	2	M_01
TP_02	Budowa wewnętrzna i sposób działania mikroprocesora 8086: jednostka arytmetyczno-logiczna, rejestry wewnętrzne i współpraca z pamięcią zewnętrzną, magistrale adresowa, danych i sterująca. Segmentacja i adresowanie pamięci.	wykład podający	2	M_01
TP_03	Podstawy asemblera dla procesora Intel 8086, podstawowe dyrektywy, instrukcje arytmetyczne, deklaracje zmiennych i przesyłanie danych. Narzędzia do tworzenia programów: asembler, linker i debugger w środowisku DOS. Przykład tworzenia i uruchamiania prostego programu.	wykład podający	4	M_01
TP_04	Przegląd instrukcji asemblerowych: operacje logiczne i arytmetyczne, instrukcje skoków warunkowych i bezwarunkowych, definicje i wywoływanie procedur z parametrami – konwencja pakietu BORLAND C. Przykłady programów. <i>Review of assembly instructions: logical and arithmetic operations, conditional and unconditional jump, definitions and calling procedures with parameters - BORLAND C package convention of use. Examples of programs.</i>	wykład podający	4	M_01
TP_05	Przerwania programowe – funkcje BIOSu i DOSu – przykłady praktycznego użycia. Rola tablicy wektorów przerw w standaryzacji systemu operacyjnego.	wykład podający	2	M_01

TP_06	<p>Karta graficzna – praca w trybie graficznym i tekstowym. Przykłady programów wykorzystujących bezpośredni dostęp do pamięci ekranu i funkcje BIOSu. Omówienie złożoności czasowej algorytmów graficznych. Współczesne karty graficzne i magistrale, złącza, okablowanie. Wykorzystanie procesorów i pamięci karty graficznej przez zaawansowane programy graficzne – wiadomości podstawowe.</p>	wykład podający	2	M_01
TP_07	<p>System przerwania sprzętowych – zgłaszanie i wykonywanie obsługi przerwania. Zadania kontrolera I8259A – maskowanie i rozstrzygnięcie priorytetu. Współczesne rozwiązania systemu przerwania sprzętowych w innych procesorach.</p>	wykład podający	2	M_01
TP_08	<p>Zegar systemowy i zegar czasu rzeczywistego w systemie DOS. Rola pamięci nieulotnej typu NV-RAM. Przechowywanie konfiguracji komputera w pamięci z kontrolą sumy kontrolnej. Współczesne źródła wskazujące dokładny czas.</p>	wykład podający	2	M_01
TP_09	<p>Klawiatura i jej działanie. Sterownik klawiatury, przerwanie sprzętowe INT 8H i programowe (BIOSu) INT 16H. Rola bufora klawiatury i sposoby kodowania znaków. Realizacja transmisji danych. Realizacja klawiatur przemysłowych oraz dla graczy.</p>	wykład podający	2	M_01
TP_10	<p>Porty równoległe i szeregowy. Ramka znaku, standardy RS-232C i RS-485 – zastosowania praktyczne. Problemy związane z transmisją: zakłócenia, konieczność stosowania izolacji galwanicznej. Zastosowanie interfejsów w przemyśle – wiadomości podstawowe.</p>	wykład podający	2	M_01

TP_11	Tendencje rozwojowe w komputerach klasy PC. Budowa i sposób działania mikroprocesora wielordzeniowego. Płyty główne i współczesne układy peryferyjne. Magistrale komunikacyjne wewnętrzne i zewnętrzne. Współczesne nośniki danych. Porównywanie wydajności układów komputerowych. Podstawowe informacje o wielozadaniowych systemach operacyjnych i dostępu do zasobów sprzętowych komputera.	wykład podający	6	M_02
		laboratorium		
TP_12	Podstawy obsługi programu asemblera i debuggera symbolicznego procesora Intel 8086 z pakietu Borland C 3.1 w środowisku symulatora DOSBOX. Zapoznanie z opcjami asemblacji, linkowania i uruchamiania programów w środowisku debugera.	praca indywidualna	2	M_03, M_04
TP_13	Deklarowanie zmiennych różnych typów w asemblerze, wartości początkowe, przydzielanie adresów przez linker, tworzenie prostego programu asemblerowego, wywoływanie funkcji DOS (powrót do systemu operacyjnego), prawidłowe dokumentowanie programów.	praca indywidualna	4	M_03, M_04
TP_14	Podstawowe instrukcje asemblerowe: kopiowanie danych przy zastosowaniu różnych sposobów adresowania, zastosowanie instrukcji arytmetycznych i logicznych, rola znaczników słowa stanu.	praca indywidualna	4	M_03, M_04
TP_15	Realizacja instrukcji wyboru <i>if...then, if...then...else</i> przy zastosowaniu instrukcji skoków warunkowych i bezwarunkowych.	praca indywidualna	4	M_03, M_04
TP_16	Realizacja instrukcji pętli programowych <i>do...while</i> i <i>while...do</i> . Optymalizacja programu, użycie rejestrów.	praca indywidualna	4	M_04, M_05

TP_17	Instrukcje obsługi stosu, tworzenie podprogramów, sposoby przekazywania parametrów. Budowanie funkcji rekurencyjnych.	praca indywidualna	4	M_04
TP_18	Wywoływanie funkcji systemowych BIOSu (dostęp do ekranu w trybie tekstowym i graficznym), realizacja programów z „ominięciem” BIOS-u. Określenie złożoności obliczeniowej tych dwóch sposobów. Dyskusja o znaczeniu standaryzacji w systemach komputerowych.	praca indywidualna	4	M_04
TP_19	Zaawansowane sposoby optymalizacji programów assemblerowych, wyszukiwanie najczęściej wykonywanych fragmentów kodu, używanie rejestrów roboczych, organizacja funkcji obsługujących przerwania sprzętowe.	praca indywidualna	4	M_05

ZALECANA LITERATURA (w tym pozycje w języku obcym)

Literatura podstawowa (powinna być dostępna dla studenta w uczelnianej bibliotece):

1. P. Metzger: *Anatomia PC*, Wyd. VIII, Helion Gliwice 2003 i nowsze
2. O. Antemijczuk: *Asembler: ćwiczenia praktyczne*, Helion, Gliwice 2002
3. W. Stallings: *Organizacja i architektura systemu komputerowego*, WNT Warszawa 2002
4. L. Null, J. Lobur: *Struktura organizacyjna i architektura systemów komputerowych*, Helion Gliwice 2004
5. B. Taylor: *BIOS dla programujących w językach C i C++*, Nakom, Poznań, 1999

Literatura uzupełniająca:

1. V.G. Oklobdzija: *The Computer Engineering handbook*, CRC Press, 2002
2. J. Kalisz: *Podstawy elektroniki cyfrowej*, WKiŁ Warszawa 2004
3. A. Tanenbaum: *Strukturalna organizacja systemów komputerowych*, Helion Gliwice 2006
4. A. Sorupski: *Podstawy budowy i działania komputerów*, WKiŁ Warszawa, 1997

III. INFORMACJE DODATKOWE

Odniesienie efektów uczenia się określonych dla zajęć i treści programowych do form zajęć i metod oceniania

Symbol efektu uczenia się określonego dla zajęć	Symbol treści programowych realizowanych w trakcie zajęć	Formy zajęć i metody dydaktyczne prowadzenia zajęć umożliwiające osiągnięcie założonych efektów uczenia się *	Metody weryfikacji osiągnięcia efektów uczenia się przypisanych do zajęć #
	Wiedza	wykład	
M_01	TP_01 - TP_10	wykład podający	zaliczenie pisemne
M_02	TP_11	wykład podający	zaliczenie pisemne
	Umiejętności	ćwiczenia, laboratorium, projekt, zajęcia praktyczne	

M_03	TP_12, TP_13, TP_14, TP_15	praca indywidualna	prezentacja efektów pracy
M_04	TP_12 - TP_18	praca indywidualna	prezentacja efektów pracy
M_05	TP_16, TP_19	praca indywidualna	prezentacja efektów pracy
	Kompetencje społeczne	ćwiczenia, laboratorium, projekt, zajęcia praktyczne	
...			

Metody weryfikacji osiągnięcia efektów uczenia się określonych dla zajęć, powinny być zróżnicowane w zależności od kategorii, tj. inne dla kategorii wiedza i inne dla kategorii umiejętności i kompetencje społeczne.

Dla wykładu:

* np. wykład podający, wykład problemowy, ćwiczenia oparte na wykorzystaniu różnych źródeł wiedzy

np. egzamin ustny, test, prezentacja, projekt

Zaleca się podanie przykładowych zadań (pytań) służących weryfikacji osiągnięcia efektów uczenia się określonych dla zajęć.

BILANS PUNKTÓW ECTS

OBCIĄŻENIE PRACĄ STUDENTA (godziny)

Forma aktywności	Liczba godzin *
Godziny zajęć (według harmonogramu) z nauczycielem akademickim lub inną osobą prowadzącą zajęcia	60
Praca własna studenta	25
SUMA GODZIN:	85

OBCIĄŻENIE PRACĄ STUDENTA (punkty ECTS)

		Liczba punktów ECTS	
SUMARYCZNA LICZBA PUNKTÓW ECTS PRZYPIŚCIANYCH DO ZAJĘĆ	Praca studenta wymagająca bezpośredniego kontaktu z nauczycielem akademickim lub inną osobą prowadzącą zajęcia	Ogółem: 3	2
	Praca własna studenta		1

* godziny lekcyjne, czyli 1 godz. oznacza 45 min;

OPIS PRACY WŁASNEJ STUDENTA:

Praca własna studenta musi być precyzyjnie opisana, uwzględniając charakter praktyczny zajęć. Należy podać symbol efektu uczenia się, którego praca własna dotyczy oraz metody weryfikacji efektów uczenia się stosowane w ramach pracy własnej.

Przykładowe formy aktywności: (1) przygotowanie do zajęć, (2) opracowanie wyników, (3) czytanie wskazanej literatury, (4) napisanie raportu z zajęć, (5) przygotowanie do egzaminu, opracowanie projektu.

Praca własna studenta obejmuje:

1. W zakresie zajęć laboratoryjnych przygotowywanie się do zajęć na podstawie podanej literatury (ok.10 godzin).

Weryfikacja pracy własnej w formie elementarnych pytań zadawanych na początku zajęć dotyczących zagadnień będą realizowane na bieżących zajęciach.

2. Przygotowywanie się do egzaminu i studiowanie zalecanej literatury (ok. 15 godzin). Weryfikacja pracy własnej w formie zaliczenia pisemnego.

KRYTERIA OCENIANIA

Ocena kształtująca:

Elementami oceny kształtującej podczas wykładu są krótkie pytania zwykle na początku zajęć dotyczące kluczowych zagadnień i pojęć z zajęć poprzednich. W trakcie zajęć wykładowca upewnia się, czy studenci posiadają wystarczającą wiedzę z innych przedmiotów (np. podstaw programowania, logiki matematycznej) przypominając ją pokrótce.

W ramach laboratorium postępy w realizacji pracy indywidualnej studenta wykonywane są na bieżąco podczas zajęć. Na początku zajęć prowadzący zadaje pytania w zakresie wiedzy potrzebnej do realizacji zadania. Możliwe jest wówczas przypomnienie wiadomości lub wyjaśnienie wątpliwości.

Ocena podsumowująca:

Ocena opanowania materiału nauczania w formie zaliczenia pisemnego obejmuje wiedzę z zakresu:

- elementów budowy komputera i ich działania,
- organizacji systemu DOS i BIOS,
- podstaw stosowania języka assemblera,
- systemu przerwania,
- budowy współczesnych komputerów i tendencji rozwojowych.

Ocenę niedostateczną otrzymuje student który nie zna podstawowych pojęć i nie umie wytłumaczyć sposobu funkcjonowania oraz programowania niskopoziomowego..

Na ocenę dostateczną wystarczy wykazać się znajomością podstawowych pojęć z zakresu budowy komputera, wiedzy z zakresu systemu DOS i BIOS oraz znać podstawowe instrukcje assemblerowe. Uzyskanie oceny dobrej lub bardzo dobrej wymaga dodatkowo szczegółowej znajomości zagadnień takich jak: system przerwania (cykl zgłoszenia i obsługi przerwania), budowa i obsługa układów peryferyjnych (zegar, klawiatura, porty), znaczenie systemu BIOS i DOS, magistrale, budowa wewnętrzna i cykl rozkazowy mikroprocesora, parametry współczesnych komponentów komputera (w tym kart graficznych).

W zakresie zajęć laboratoryjnych weryfikowane są przede wszystkim umiejętności. Składają się na nie trzy zasadnicze elementy:

- posługiwanie się emulatorem i środowiskiem programistycznym,
- konstruowanie prostych programów w assemblerze,
- wywoływanie funkcji BIOS/DOS,
- realizacja bardziej złożonych programów z różnymi modelami i trybami adresowania pamięci.

Ocenę niedostateczną otrzymuje student, który nie potrafi zrealizować żadnego programu. Na ocenę dostateczną wymagane jest Zrealizowanie i uruchomienie prostego programu (nawet kilkulinijkowego). Na ocenę dobrą wymagana jest umiejętność użycia funkcji BIOSu/DOSu i realizacja podstawowych instrukcji strukturalnych. Na ocenę bardzo dobrą student musi umieć realizować pętle i podprogramy w języku assemblera.

INFORMACJA O PRZEWIDYWANEJ MOŻLIWOŚCI WYKORZYSTANIA B-LEARNINGU
nie przewiduje się

INFORMACJA O PRZEWIDYWANEJ MOŻLIWOŚCI WYKORZYSTANIA E-LEARNINGU
nie przewiduje się

.....
(data, podpis Koordynatora
odpowiedzialnego za zajęcia)

.....
(data, podpis Dyrektora Instytutu/
Kierownika Jednostki Międzyinstytutowej)

.....

(data, podpis Kierownika Zakładu)

*Uwaga:
Karta opisu zajęć (sylabus) musi być dostępna dla studenta.*