



Kierunek studiów	Informatyka
Profil	Praktyczny
Stopień studiów	1-go stopnia
Forma studiów	niestacjonarne

Sylabus przedmiotu Niskopoziomowa analiza kodu

1. Dane podstawowe

Status programowy przedmiotu	Blok B: Przedmioty unikatowe dla Inżynierii Oprogramowania
Rodzaj przedmiotu	Fakultatywny
Kod przedmiotu	IZ-NAK-ZR
Rok studiów	4
Semestr	8
Osoba odpowiedzialna za przedmiot	dr inż. Adam Patkowski
Język wykładowy	polski

2. Wymiar godzin i forma zajęć

Rodzaj	Liczba godzin
Wykład	14
Laboratorium	16
Razem godzin	30

3. Cele przedmiotu

Kod	Cel
CP1	Znajomość języka wewnętrznego mikroprocesorów Intel 80x86 (16, 32 i 64-bitowych). Znajomość podstawy asemblera i jego użycia w programowaniu.
CP2	Znajomość zasad komunikacji programów z systemem operacyjnym, funkcji API i wykorzystania pamięci. Znajomość budowy modułów ładowalnych EXE i Portable Executable. Znajomość budowy i zasad działania programów sterowanych zdarzeniami. Znajomość i rozumienie podstawowej metodyki inspekcji kodu.
CP3	Umiejętność wykonania inspekcji kodu modułów ładowalnych i dokonywanie ich prostych modyfikacji.
CP4	Umiejętność tworzenia prostych konstrukcji w asemblerze na potrzeby modyfikacji kodu i użycia ich INLINE, np. w C.

4. Wymagania wstępne w zakresie wiedzy, umiejętności i innych kompetencji

Znajomość architektury komputerów. Ogólna orientacja w budowie stacji roboczej PC. Znajomość podstaw systemu operacyjnego Windows. Znajomość podstaw algorytmów. Podstawowa znajomość języka C.

5. Efekty uczenia się

Wiedza

Kod	Student zna i rozumie:	Realizuje cel	Efekty kierunkowe
EU-W1	nowe mikroprocesory linii x86 i elementy ich historii, architekturę i podstawy organizacji wewnętrznej.	CP1	IK6_W14, IK6_W21
EU-W2	reguły języka wewnętrznego mikroprocesorów x86 i x64 oraz wybrane instrukcje tego języka („rozkazy”) dla trybów poza SMM.	CP1, CP4	IK6_W14, IK6_W21
EU-W3	komunikację aplikacji z systemami Windows oraz funkcje API. Zna podstawowe struktury plików modułów ładowalnych (EXE i PE) oraz użycie pamięci przez aplikacje i biblioteki DLL. Zna konstrukcje programów sterowanych zdarzeniami.	CP2	IK6_W14, IK6_W21
EU-W4	podstawowe metodyki inspekcji kodu. Zna zasady działania i obsługi reprezentatywnych narzędzi tych metodyk.	CP3	IK6_W14, IK6_W21

Umiejętności

Kod	Student potrafi:	Realizuje cel	Efekty kierunkowe
EU-U1	wykonać inspekcję kodu programów (z pliku i w pamięci).	CP3	IK6_U07, IK6_U25
EU-U2	dokonać prostych modyfikacji kodu programu dla osiągnięcia pożądanego zachowania się.	CP3	IK6_U07, IK6_U25
EU-U3	tworzyć proste konstrukcje w zapisie assemblerowym i użyć ich IN-LINE w języku wysokopoziomowym.	CP4	IK6_U07, IK6_U21

Kompetencje

Kod	Student jest gotów do:	Realizuje cel	Efekty kierunkowe
EU-K1	zrozumienia dezaktualizacji wiedzy. Rozumie potrzebę ciągłej pogoni intelektualnej i rozwoju zawodowego za zmianami technologii.	CP1, CP2	IK6_K01, IK6_K02, IK6_K03
EU-K2	zrozumienia groźby przekroczenia dopuszczalnych (prawem i etyką zawodową) granic wykorzystania nabytej wiedzy.	CP3	IK6_K01, IK6_K04

6. Treści programowe

Kod	Tematyka	wykład	laboratorium	Realizuje efekt
TP1	Architektury nowych mikroprocesorów linii 80x86 (16, 32 i 64-bitowych)	2	0	EU-K1, EU-W1
TP2	Język wewnętrzny procesorów x86, x64. Wybrane rozkazy. Tryby 16, 32 i 64-bitowe.	6	4	EU-K1, EU-W2
TP3	Reverse engineering, narzędzia i zasady inspekcji kodu binarnego.	0	8	EU-K2, EU-U1, EU-U2, EU-W4
TP4	Aplikacje Windows, format PE. Programy sterowane zdarzeniami. Komunikacja z systemem.	2	2	EU-K1, EU-U1, EU-W3
TP5	Wynik kompilacji w języku wysokopoziomowym. Łączenie z assemblerem.	2	0	EU-U3
TP6	Organizacja wewnętrzna systemu Windows. Funkcje API. Zaawansowane narzędzia.	2	2	EU-K2, EU-U1, EU-U2, EU-W3, EU-W4

Razem godzin: 30

7. Metody kształcenia

Kod	Metoda
MK1	Wykład z prezentacją komputerową
MK2	Zadania warsztatowe w laboratorium - krokowe rozwiązywanie zadań z określaniem kolejnych celów przez prowadzącego i burzę mózgów oraz dyskusją proponowanych rozwiązań.

8. Nakład pracy studenta

Aktywność studenta	Obciążenie
Przygotowanie do laboratorium	45
Praca z nauczycielem związana z: laboratorium	16
Praca z nauczycielem związana z: wykład	14
Liczba punktów ECTS (1 punkt=25h)	3
Procentowy udział pracy własnej studenta w sumarycznym obciążeniu studenta	60,00%
Sumaryczne obciążenie pracą studenta	75

9. Status zaliczenia przedmiotu

Ocena końcowa na podstawie liczby punktów zgromadzonych na laboratoriach. Za obecność na każdej godzinie (bez spóźnień) student otrzymuje 1 punkt, nie więcej jednak niż 14 punktów łącznie. Za każdą wypowiedź lub odpowiedź na pytanie student otrzymuje od -1 do +1 pkt. według arbitralnej oceny prowadzącego. Ocena końcową wystawia prowadzący, uwzględniając liczbę punktów studenta. Orientacyjne minimalne wartości dla poszczególnych ocen to: DST 14; DB 18; BDB 22.

Forma studiów	Egzamin	Praca egzaminacyjna	Zaliczenie	Praca zaliczeniowa
niestacjonarne			×	

10. Metody weryfikacji efektów uczenia się

Składowe oceny końcowej

Forma sprawdzenia	Wybrana forma	Punktacja	Realizuje efekt
Egzamin pisemny			
Egzamin ustny			
Sprawdzian pisemny			
Zaliczeniowy przegląd prac			
Referat pisemny			
Referat ustny			
Kolokwium			
Praca domowa			
Miniprojekt			
Praca na zajęciach	×	52	EU-U3, EU-U2, EU-U1, EU-K1
Projekt z dokumentacją			
Ustna prezentacja projektu			
Obecność na zajęciach	×	48	EU-W4, EU-W3, EU-W2, EU-W1, EU-K2
Sprawdzian ustny			
Kartkówka			
Aktywność na zajęciach			
Egzaminacyjny przegląd prac			
Sprawozdanie z praktyki zawodowej			
Prezentacja indywidualna			
Prezentacja zespołowa			

Zasady wyliczania oceny z przedmiotu

Zakres punktów	Ocena
0 – 13	2,0
14 – 15	3,0
16 – 17	3,5
18 – 19	4,0
20 – 21	4,5
22 – 28	5,0

11. Macierz realizacji przedmiotu

Efekt uczenia się	Cel przedmiotu	Treści programowe	Metody kształcenia
EU-W1	CP1	TP1	MK1, MK2
EU-W2	CP1, CP4	TP2	MK1, MK2
EU-W3	CP2	TP4, TP6	MK1, MK2
EU-W4	CP3	TP3, TP6	MK1, MK2
EU-U1	CP3	TP3, TP4, TP6	MK1, MK2
EU-U2	CP3	TP3, TP6	MK1, MK2
EU-U3	CP4	TP5	MK1, MK2
EU-K1	CP1, CP2	TP1, TP2, TP4	MK1, MK2
EU-K2	CP3	TP3, TP6	MK1, MK2

12. Odniesienie efektów uczenia się

Efekt uczenia się	Efekty kształcenia dla kierunku studiów	Charakterystyki drugiego stopnia w obszarze kształcenia
EU-W1	IK6_W21, IK6_W14	P6S_WG
EU-W2	IK6_W21, IK6_W14	P6S_WG
EU-W3	IK6_W21, IK6_W14	P6S_WG
EU-W4	IK6_W21, IK6_W14	P6S_WG
EU-U1	IK6_U25, IK6_U07	P6S_UW
EU-U2	IK6_U25, IK6_U07	P6S_UW
EU-U3	IK6_U21, IK6_U07	P6S_UW
EU-K1	IK6_K03, IK6_K02, IK6_K01	P6S_KK
EU-K2	IK6_K04, IK6_K01	P6S_KK, P6S_KO

13. Literatura

Literatura podstawowa

1. Gary Syck, Turbo Assembler. Biblia użytkownika., LTP Oficyna Wydawnicza, 2002
2. Intel, Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 1: Basic Architecture, Intel, 2020
3. Intel, Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 2 (2A, 2B & 2C): Instruction Set Reference, A-Z, Intel, 2020

Literatura uzupełniająca

1. Randall Hyde, Art of Assembly Language Programming, 2003

Strony WWW

1. Webster The Place on the Internet to Learn Assembly, <http://www.plantation-productions.com/Webster/>
2. Debugger IDA, <https://www.hex-rays.com/index.shtml>
3. OllyDbg, <http://www.ollydbg.de/>
4. AMD, Dokumentacja AMD, <http://developer.amd.com/resources/documentation-articles/developer-guides-manuals/>
5. Intel, Dokumentacja procesorów Intela, <http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html>

Pozostałe

1. Aktualne materiały z Internetu: właściwe tematycznie FAQs, crackmes, tutoriale, dokumentacje oprogramowania.

14. Informacje o nauczycielach akademickich

Osoby odpowiedzialne za przedmiot

1. dr inż. Adam Patkowski

Osoby prowadzące przedmiot

1. dr inż. Adam Patkowski