

HỢP NGỮ VÀ DỊCH NGƯỢC
(ASSEMBLY PROGRAMMING AND REVERSE ENGINEERING)
Đề cương chi tiết (Course Syllabus)

1. General Information

Course name: Hợp ngữ và Dịch ngược (Assembly Programming and Reverse Engineering)

Course code: SEC1302_CLC

Course type: Compulsory

Number of credits: 3

2. Objectives

Knowledge:

The course introduces the x86 architecture, basic knowledge of assembly language programming, and fundamental tools and techniques of reverse engineering.

Skills:

On successful completion of this course a student will be able to:

- Create, compile and analyze x86 programs;
- Apply reversed tools and techniques to analyze x86 specific programs in order to understand their functionality and behaviors.

Attitude:

Students are required to attend the classes and complete assignments/projects.

3. Abstracts

The course enables learners to explore the world of low-level programming and x86 computer architecture. Assembly language is a powerful tool that allows learners to write programs that interact directly with the hardware, providing learners with the insight of computers operations.

Moreover, the course presents learners with tools and techniques of reverse engineering, enabling learners to comprehend the design, functionality, and behavior of software programs without access to their source code. This skill is essential for cybersecurity, software development, and maintaining legacy systems.

4. Teaching and learning methods

Lectures: 24h

Exercises: 5h

Projects: 8h

Labs: 8h

Individual reading: 0h

5. Prerequisites: None

6. Learning outcomes

After completing this course, the student is able to:

[CLO1] Describe the tools and techniques for compiling and analyzing basic assembly programs, and for reverse engineering.

[CLO2] Apply tools and techniques to write, compile, and analyze x86 programs to solve specific problems.

[CLO3] Apply appropriate tools and techniques of reverse engineering to analyze computer programs in order to understand their functionality and behaviors.

7. Assignment criteria

Learning outcomes	Assignment criteria
[CLO1] Describe the tools and techniques for compiling and analyzing basic assembly programs, and for reverse engineering.	Chapter 1, Chapter 2, Chapter 4
[CLO2] Apply tools and techniques to write, compile, and analyze x86 programs to solve specific problems.	Chapter 2, Chapter 3, Chapter 5
[CLO3] Apply appropriate tools and techniques of reverse engineering to analyze computer programs in order to understand their functionality and behaviors.	Chapter 4, Chapter 5

8. Outlines

Chapter 1 Introduction to Assembly Language

- 1.1. History and Evolution
- 1.2. Basic Concepts and Terminology
- 1.3. Assembly Language vs. High-Level Languages
- 1.4. Processor Architecture
- 1.5. Memory Hierarchy
- 1.6. Instruction Set Architecture (ISA 32&64)

Chapter 2 x86 Assembly Language Fundamentals

- 2.1. Data Representation
- 2.2. Binary, Hexadecimal, and Decimal Systems
- 2.3. Basic Instructions (MOV, ADD, SUB, etc.)
- 2.4. Extended Instructions
- 2.5. Addressing Modes

Chapter 3 Programming in Assembly Language

- 3.1. Writing Simple Programs
- 3.2. Control Flow (Jumps, Loops, and Conditional Statements)
- 3.3. Procedures and Functions
- 3.4. Stack Operations
- 3.5. Interrupts and I/O Operations

3.6. Practical applications

Chapter 4. Introduction to Reverse Engineering

- 4.1. Definition and Importance
- 4.2. Applications in Security and Forensics
- 4.3. Legal and Ethical Considerations
- 4.4. Tools and Techniques

Chapter 5. Software Reverse Engineering

- 5.1. Static Analysis
- 5.2. Dynamic Analysis
- 5.3. Binary Analysis
- 5.4. Reverse Engineering Automation
- 5.5. Case studies

9. Required Textbooks

- [1] Kip Irvine, Assembly Language for x86 Processors, 8th edition, Prentice-Hall (Pearson Education), June 2019, ISBN: 978-0135381656

10. Suggested Textbooks

- [2] Jeff Duntemann, Assembly Language Step-By-Step, Wiley Publishing, Inc., ISBN: 978-0-470-49702-9, 2011
- [3] Randal E. Bryant and David R. O'Hallaron, Computer Systems: A Programmer's Perspective, 3/E (CS:APP3e), Pearson; 3rd edition, ISBN-13 : 978-0134092669, 2023
- [4] Reginald Wong, Mastering Reverse Engineering, 2018, Publisher(s): Packt Publishing, ISBN: 9781788838849

11. Schedule

Main contents	Duration	Specific contents
Chapter 1 Introduction to Assembly Language	3h lecture	<ul style="list-style-type: none">– History and Evolution– Basic Concepts and Terminology– Assembly Language vs. High-Level Languages– Processor Architecture– Memory Hierarchy– Instruction Set Architecture (ISA 32 & 64)
Chapter 2 x86 Assembly Language Fundamentals	3h lecture 1h exercise	<ul style="list-style-type: none">– Data Representation– Binary, Hexadecimal, and Decimal Systems– Basic Instructions (MOV, ADD, SUB, etc.)– Extended Instructions– Addressing Modes

Chapter 3 Programming in Assembly Language	6h lecture 2h exercise 3h project 3h lab	<ul style="list-style-type: none"> – Writing Simple Programs – Control Flow (Jumps, Loops, and Conditional Statements) – Procedures and Functions – Stack Operations – Interrupts and I/O Operations – Practical applications
Chapter 4. Introduction to Reverse Engineering	6h lecture 1h exercise 3h project 3h lab	<ul style="list-style-type: none"> – Definition and Importance – Applications in Security and Forensics – Legal and Ethical Considerations – Tools and Techniques
Chapter 5. Software Reverse Engineering	6h lecture 1h exercise 2h project 2h lab	<ul style="list-style-type: none"> – Static Analysis – Dynamic Analysis – Binary Analysis – Reverse Engineering Automation – Case studies

12. Grading Policy

Attendance:	10%
Mid-term exam/exercises:	10%
Course projects:	30%
Final examination:	50%