VIETNAM NATIONAL UNIVERSITY – HO CHI MINH CITY

UNIVERSITY OF INFORMATION TECHNOLOGY

SYLLABUS
NT209 - COMPUTER SYSTEM PROGRAMMING

1. GENERAL INFORMATION

Course name (Vietnamese):	**Lập trình hệ thống**
Course name (English):	**Computer System programming**
Code:	NT209
Type of course:	Fundamental
Department:	Faculty of Computer Networks and Communications
Instructor:	PhD. Pham Van Hau, MSc. Do Thi Huong Lan, MSc. Do Thi Thu Hien Email: hiendtt@uit.edu.vn
Number of credits:	3
Theory:	2
Lab:	1
Self-study:	
Prerequisite course(s):	
Pre-course(s):	IT001 - Introduction to programming IT006 - Computer architecture

2. COURSE DESCRIPTION

This course aims to provide students with fundamental skills and knowledge of:

- Basic concepts of computer-level programming in assembly language: data type, registers, memory, instructions... to understand how a program in a high-level programming language can be presented at the computer level and vice versa.
- Basic knowledge and skills in program optimization, concepts of stack, pointer, cache, and other computer system elements... to offer students the ability to build programs with better performance.
- Basic knowledge and skills in reverse engineering, buffer overflow attack, and program debugging.

3. COURSE GOALS

Table 1.

Goal No.	**Goal description**	**Program outcomes**	**Level** (Bloom)
G1	Describe the functions of components and data presentation in a computer system	*L02 (2.1, 2.2)*	*Knowledge - 3*
G2	Describe the functions of assembly instruction in the operation of programs	*L02 (2.1, 2.2, 2.4)*	*Knowledge - 3*
G3	Translate high-level language based programs to assembly-based presentation and vice versa, intepret assembly code to understand or exploit the operation of programs.	*L02 (2.7.2), L03 (3.1, 3.2)*	*Skill - 3*

4. COURSE LEARNING OUTCOMES

Table 2.

Course outcomes	**Descriptions**	**Level of teaching**
G1.1	Describe the functions of components in a computer system	*I*
G1.2	Describe data presentation in a computer system	*I, T*
G2.1	Describe the functions of assembly instruction in the operation of programs	*I, T*
G3.1	Translate high-level language based programs to assembly-based presentation and vice versa	*I, T, U*
G3.2	Intepret assembly-based program to discover or exploit the operation of program	*I, T, U*

(I – Introduce, T – Teach, U – Utilize)

5. COURSE CONTENT, LESSON PLAN (Nội dung môn học, kế hoạch giảng dạy)

a. Theory

Table 3.

Week (1.5 hours)	Contents	Course learning outcomes	Activities	Assessment element
1 (1.5 class hours)	**Chapter 1: Introduction to Computer system programming** - Overview of real-world facts that needs the knowledge of computer system programming. - Overview of course goals, contents - Assessment	*G1.1*	**Teaching**: lecturer gives instructions, question **Study in class**: exchange related issues, problems. **Self-study**: do homework	A1, A2
2 (1.5 class hours)	**Chapter 2: Data presentation and manipulation** - Data storing • The base of binary, octal, hexan, decimal • Words, data size • Bitwise operations, shifts - Integer presentation • Byte ordering: little/big endian • Signed and unsigned integer • Expand and contract numbers - Arithmetic operations • Unsigned addition • Multiplication, Division - Floating point • IEEE Floating point 754 • Floating point arithmetic operations	*G1.2*	**Teaching**: lecturer gives instructions, question, demo **Study in class**: exchange related issues, problems. **Self-study**: do homework	A1, A2
3, 4 (3 class hours)	**Chapter 3.1: Basic to computer-level programming – Assembly** - Introduction to IA32 and x86_64 - Fundamental concepts in computer system programming: registers, memory,... - Assembly-based operations: • Instructions • Operand types: registers, memory, immediates • Data moving, memory addressing	*G1.1, G1.2, G2.1, G3.1*	**Teaching**: lecturer gives instructions, question, demo, practice exercises with detailed instructions or quizzes **Study in class**: exchange related issues, problems, do in-class exercises/quizzes	A1, A2

	● Two operand and one operand arithmetic instructions - Exercises in basic computer system programming: understanding assembly instructions, operands...		**Self-study**: do homework	
5, 6 (3 class hours)	**Chapter 3.2: Machine-level control** - Condition codes - Application of condition code in controlling program flows: ● Conditional branches, jump ● Loops ● Switch - Exercises in machine-level control: understand assembly presentation of if/else, loops, switch statements...	*G1.1, G2.1, G3.1*	**Teaching**: lecturer gives instructions, question, demo, practice exercises with detailed instructions or quizzes **Study in class**: exchange related issues, problems, do in-class exercises **Self-study**: do homework	A1, A2
7 (1.5 class hours)	**Chapter 3.3: Procedure** - Fundamental concepts in procedure calling in IA32: stack, return address,... - Procedure operations: ● Control flow from caller to callee and vice versa ● Data passing: arguments, return values... ● Recursive procedure example **Mid-term examination preparation**	*G1.1, G2.1, G3.1*	**Teaching**: lecturer gives instructions, question, demo, practice exercises with detailed instructions or quizzes **Study in class**: exchange related issues, problems, do in-class exercises **Self-study**: do homework, prepare for mid-term exam	A2, A3
8 (1.5 class hours)	**Chapter 3.3: Procedure (cont)** - Procedure calling in x86_64 compared to IA32. - Exercises in procedure calling. - Introduction to reverse engineering.	*G1.1, G2.1, G3.1, G3.2*	**Teaching**: lecturer gives instructions, question, demo, practice exercises with detailed instructions or quizzes **Study in class**: exchange related	A2, A3

			issues, problems, do in-class exercises **Self-study**: do homework	
9 (1.5 class hours)	**Chapter 3.4: Advanced data presentation** - Array: one-dimensional, nested, multi-level arrays - Structure: allocation, access and alignment in struct - Union - Excercises	*G1.1, G2.1, G3.1*	**Teaching**: lecturer gives instructions, question, demo, practice exercises with detailed instructions or quizzes **Study in class**: exchange related issues, problems, do in-class exercises **Self-study**: do homework	A2, A3
10, 11 (3 class hours)	**Chapter 3.5: Introduction to buffer overflow** - Fundametal knowledge of buffer overflow: • Vulnerability • Protection - Exercise of exploiting buffer overflow vulnerability in executable files.	*G2.1, G3.1, G3.3, G3.2*	**Teaching**: lecturer gives instructions, question, demo, in-class buffer overflow exploiting exercise **Study in class**: exchange related issues, problems, do in-class exercises **Self-study**: do homework	A2, A3
12 (1.5 class hours)	**Chapter 4: Linking** - Static linking - Object file and ELF format - Symbol and Symbol table - Linker operations: • Symbol resolution • Relocation - Dynamic Linking with Shared Object files	*G1.1, G2.1*	**Teaching**: lecturer gives instructions, question, demo **Study in class**: exchange related issues, problems **Self-study**: do homework	A3
13 (1.5 class hours)	**Chapter 5: Memory hierarchy and Cache** - Memory hierarchy - Cache memory	*G1.1, G2.1*	**Teaching**: lecturer gives instructions, question, demo	A3

			Study in class: exchange related issues, problems **Self-study**: do homework	
14&15 (1.5 class hours)	**Course review and final exam preparation**	*G1-G3*	**Teaching**: lecturer gives instructions, course review, quizzes... **Study in class**: exchange related issues, problems, do in-class quizzes **Self-study**: prepare final examination	A3

b. Lab

Table 4.

Session (4.5 hours)	**Contents**	**Course learning outcomes**	**Activities**	**Assessment element**
1	Lab 1: Data Lab	G1.2	**Teaching:** lecturer describes the objective of the lab and gives instructions for students. **Study in class:** Students follow the instruction of the lab, report in-class requirements **Self-study:** Students read the instruction and prepare the lab at home.	A2
2	Lab 2: Basic assembly programming	G2.1	**Teaching:** lecturer describes the objective of the lab and gives instructions for students. **Study in class:** Students follow the instruction of the lab, report in-class requirements. **Self-study:** Students read the instruction and prepare the lab at home.	A2
3	Lab 3: Basic reverse engineering	G2.1, G3.1	**Teaching**: lecturer describes the objective of the lab and gives instructions for students. **Study in class**: Students follow the instruction of the lab, report in-class requirements.	A2

			Self-study: Students read the instruction and prepare the lab at home.	
4	Lab 4: Bomb Lab (advanced reverse engineering)	G2.1, G3.1	**Teaching**: lecturer describes the objective of the lab and gives instructions for students. **Study in class**: Students follow the instruction of the lab, report in-class requirements. **Self-study**: Students read the instruction and prepare the lab at home.	A2
5	Lab 5: Bufferbomb Lab (basic buffer overflow)	G2.1, G3.1, G3.2	**Teaching**: lecturer describes the objective of the lab and gives instructions for students. **Study in class**: Students follow the instruction of the lab, report in-class requirements. **Self-study**: Students read the instruction and prepare the lab at home.	A2
6	Lab 6: Bufferbomb Lab (advanced buffer overflow)	G2.1, G3.1, G3.2	**Teaching**: lecturer describes the objective of the lab and gives instructions for students . **Study in class**: Students follow the instruction of the lab, report in-class requirements. **Self-study**: Students read the instruction and prepare the lab at home.	A2

6. COURSE ASSESSMENT

Table 5.

Assessment element	Course learning outcomes	Percentage (%)
A1. Mid-term exam	G1.1, G1.2, G2.1, G3.1	*30%*
A2. Lab	G1.1, G1.2, G2.1, G3.1, G3.2	*20%*
A3. Final exam	G1.1, G1.2, G2.1, G3.1, G3.2	*50%*

a. Assessment A1

The mid-term exam includes (but is not limited to)

- Multiple choice questions: 10 - 12 questions for G1.1, G2.1
- Written: 2-3 questions for G2.1, G3.1.

b. Assessment A2

The practice test score is the average of 6 practice sessions, each practice has a marking guide.

c. Assessment A3

The final exam includes (but is not limited to)

- Multiple choice questions: 10 - 12 questions for G1.1, G1.2, G2.1
- Written: 2-3 questions for G2.1, G3.1, G3.2.

7. COURSE REQUIREMENTS AND EXPECTATIONS

- Laboratory: labs can be done in group in laboratories. Lecturer describes the objective of the lab and gives instructions for students. Students must fulfill all lecturer's requirements. Late submission is not accepted or punished according to the reason.
- Class attendance: Students are checked their attendance in class by in-class assignment. Failing to show up by the time of checking is considered to be absent.
- Students need to read slide before the class.
- Mid-term and Final examination: Students that fail to show up on the examination day without acceptable reasons will get 0.

8. COURSE MATERIALS

1. Bryant, R. E. & O'Hallaron, D. R. (2015). *A Programmer's Perspective (3rd edition)*. Addison-Wesley Publishing Company.

2. Eagle, C. (2008). *The IDA Pro Book: The Unofficial Guide to the World's Most Popular Disassembler*. No Starch Press.

3. Eilam, E. (2011). *Reversing: Secrets of Reverse Engineering*. Wiley.

9. SOFTWARE, TOOLS

1. C compiler: GCC/Online compiler (Godbolt)
2. Linux virtual machine.
3. Programming IDE for C/C++.
4. Disassembler: IDA Pro, GDB.

Date: Month, Date, Year

Faculty Head **Instructor**