Syllabus for Introduction to Engineering Computing

1 Teaching purpose

Programming embodies an abstract interaction relationship and a mode of thinking that formalizes the execution of the method. This is "computational thinking". Solving problems by writing programs and debugging codes can promote students' thinking, enhance observation and deepen the understanding of interaction relationships. Programming can enhance understanding. Writing a program is not just about solving calculations. It requires the author to think about ways to solve the problem, but also how to make the program have a better user experience, higher execution efficiency and more interesting display effects. Programming can improve efficiency, and mastering certain programming techniques will help to make better use of computers to solve tedious computing problems.

Through the study of programming courses, students' computational thinking ability and the ability to use computers to solve problems are cultivated, and students' computer skills are improved, so that students are more creative and competitive.

2 class hours

Credits: 2.0, total hours: 48 hours, including 32 hours for classroom teaching and 16 hours for experiments.

3 Course content and teaching plan

3.1 The content and plan of theoretical courses (1-16 weeks, 2 class hours/week, 32 class hours in total)

- Overview, Algorithm（2 hours）

  1. Visual Studio C Environment
  2. C program composition
  3. Header file, Data description, Function start and end flag

4. The writing format of the source code

- Data types, Operators and Expressions（2 hours）

1. C data type and its definition method

2. Types, precedence and associativity of C operators

3. Conversion and operation between different types of data

4. C expression types and evaluation rules

- Sequence structure program design (2 hours)

1. Basic sentences

2. Data input and output, call of input and output functions

3. Sequence programming

- Selection structure program design (4 hours)

1. Relational expressions, logical expressions, conditional expressions

2. if

3. switch

4. The nesting of the selection structure

Organize a class discussion, calculate and apply the branch structure, and allow students to prepare in advance.

- Loop structure (5 hours)

1.for

2.while & do while

3.continue & break

4. The nesting of the loop structur

Organize a class discussion, the application of the loop structure, and allow students to prepare in advance.

- Array (5 hours)

1. Definition, initialization and reference of one-dimensional arrays and multi-dimensional arrays

2. Strings and character arrays

- Function (4 hours)

1. Definition of function

2. The type and return value of the function

3. The transfer of formal parameters and actual parameters, parameter values

4. Function call, nested call, recursive call

5. Local variables and global variables

6. The storage category, scope and lifetime of variables

7. Internal function and external function

- Preprocessing commands (1 hour)

1. Macro definition

2. Include

- Pointer (3 hours)

1. The concept of pointer and pointer variable, pointer and address operators

2. Pointer to variable, array, string, and function, and pointer variable to variable, array, string, and function

3. Use pointer as function parameter

4. Array of pointers, pointer to pointer

- Structure and Union (2 hours)

1. The definition method and reference method of structure and union type data

2. Use pointers and structures to form a linked list, and create, output, delete and insert a singly linked list

- File (2 hours)

1. FILE type pointer

2. File opening and closing

3. File reading and writing

4. File location

- Professional application

Case analysis and explanation

3.2 Experimental teaching content and plan (9-16 weeks, 2 hours/week, 16 hours in total)

Serial number	Project	Summary of experiment content	Hours	Type (Comprehensive/Verifying/ Demonstrative)
1	The running environment of C program and the method of running a C program	Understand the basic operation method of VS compilation system, learn to use the system independently; understand how to edit, compile, connect and run a C program on the system; by running a simple C program, initially understand the characteristics of the C source program and C language program structure	2	Verification
2	Sequence structure programming experiment	Familiar with the basic data types in C language, master the methods of defining constants and variables and assigning values to them, understand the format conversion symbols used in data output; master the usage of format input/output functions; learn the design of simple sequence programs	2	Verification
3	Selection structure program design experiment	Master the use of relational operators, logical operators, and increment and decrement operators; proficiently use if and switch to write programs	2	Verification
4	Loop structure programming experiment 1	Master the use of relational operators, logical operators, increment and decrement operators; proficiently use for, while to write programs; master	2	Comprehensive

		the use of break and continue statements		
5	Loop structure program design experiment 2	Proficiency in using for, while to write programs; master the use of break and continue statements	2	Comprehensive
6	Array Experiment 1	Grasp the methods of defining one-dimensional and two-dimensional arrays; master the methods of initialization and loop assignment of one-dimensional and two-dimensional arrays; master the use mode of combining arrays and loop statements to deal with problems	2	Comprehensive
7	Array Experiment 2	Master the methods of defining one-dimensional and two-dimensional arrays; master the methods of initialization and loop assignment of one-dimensional and two-dimensional arrays; master the use mode of combining arrays and loop statements to deal with problems	2	Comprehensive
8	Function experiment	Master the method of defining functions; master the corresponding relationship between the actual parameters of the function and the value transfer rules of function calls; understand the meaning of the return value of the function, and master the correct operation of the return value of the function	2	Comprehensive