1. Sequential structure programming and C program environment

Lab Purpose

1. Master the data types of C language and understand the inherent relationship between character and integer data.

2. Master the input methods for various numerical data types

3. Learn to write and run simple program code.

4. Be further familiarized with the process of editing, compiling, connecting, and running C programs.

Lab content and steps

1、Edit, compile and run the following program. How to type in data on the keyboard for, so that a=3, b=7, y=71.82, c1='A ', c2='a'?

#include<stdio. h>

int main()

{

int a, b;

float x, y;

char c1, c2;

scanf ("a=%db=%d",&a,&b);

scanf ("%f%f",&x,&y);

scanf ("%c%c",&c1,&c2);

}

During runtime, type in data as follows, observe and analyze the output results.

A=3,b=7,x=8.5,y=71.82,A,a

2、 Write a program that translates an input string with a length of 5 into a password. The rule is to replace each letter with the fourth letter after it. For example, the fourth letter after 'A' is' E ', so 'E' substitutes 'A'. Therefore, 'China' is replaced by 'Glmre'.

1. Branch Structure Programming

Lab purpose:

Master the representation of true and false in C (0: False. non zero: True) (False=0 True=1)

Master if statement to implement selection structures

Master switch statement to implement multi-branch selection structure

Master nested selection structures

Lab content:

1. After reading x from the keyboard, calculate the value of the segmented function

1. Input months 1-12 from keyboard, output the corresponding seasons (2-4: spring, 5-7: summer, 8-10: autumn, 11-12, 1: winter)

3. loop structure programming

lab purpose:

Master the while loop, for loop, and do-while loop structures

Understand the execution process of three loop structure control statements

Accumulate experience in program debugging

lab content:

1. Enter a positive integer from the keyboard and determine if it is a prime number. Print “This is a prime” if it is a prime number, otherwise, print “This is not a prime”.

Note:

1 is not a prime.

A prime number is a natural number that can only be divided by 1 and itself.

1. Enter n numbers from the keyboard, and find the maximum number (n is entered from the keyboard)

for example, input 6 (n=6) and six numbers as 39 28 5 63 18 27, then output 63

1. ~~print the following using loop~~

~~~~

4. FUNCTIONS

lab purpose

* + 1. Master the method of defining functions
    2. Master the relationship between function arguments and formal parameters, as well as the difference between call by value and call by reference.
    3. Master the methods of nested function and recursive function calls

lab content:

1. Write a function to count characters

Requirement: When calling, a string is passed by an argument, and the class of the character is passed by an argument ( 0 indicates that the number of characters is counted, 1 indicates the number of digits is counted, 2 indicates the number of spaces is counted, and 3 indicates that the number of other characters is counted. ) The return value of the function is the counted result. The function prototype is as follows:

int count(char arr[], int class)

1. For a given positive integer which is input from the keyboard, calculate its reversed number using recursion. For example, if 1234 is input, the result should be 4321.

Requirement:

the returned value of the function should be the reversed number,

the parameters of the functions include the number to be inverted, and the reversed number

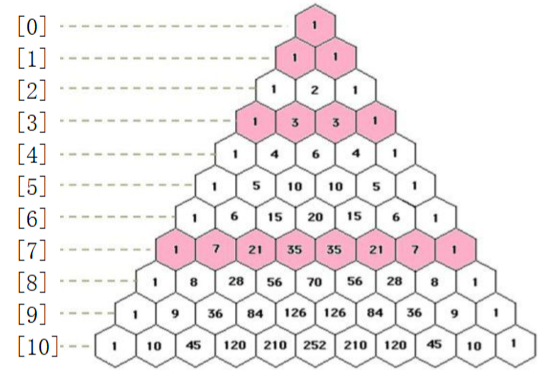
Lab 5: Arrays

lab purpose:

1. Master the definition, input and output methods of one-dimensional and two-dimensional arrays
2. Master the access methods of array members
3. Understand the use of character arrays and have a preliminary understanding of commonly used string functions
4. Mastering some classic algorithms, such as search and sorting
5. Further strengthen the program's high-level capabilities.

lab content:

1. Define an array to store 10 integers and initialize each element in the array (with no duplicate values for each element). Enter a number *num* from the keyboard and search if there is any element in the array equal to *num*. If there is, output its index. If it does not exist, output NOT FOUND!
2. Finding the sum of the main diagonals of a two-dimensional array.
3. Output the Yanghui triangular



LAB pointers and their usage

lab purpose:

(1) Master the concepts of pointers and indirect access, and be able to define and use pointer variables.

(2) Master the usage of pointer variables as function parameters

(3) Deepen understanding of how pointers and addresses are stored in memory

lab content:

1. Write a function to find the length of a string. Enter a string in the main function and output its length
2. Write a function to reverse an array using pointers. Enter the elements of an array in the main function, and pass the array name as argument.

lab 7 application of user- defined data bytes

lab purpose:

Master the definition of user-defined data types

Master the definition and usage of variables of user-defined data type

lab content:

1) define a student structure type, whose members includes student ID, name, class, score for math, score for English and score for C programming.

2) program to enter students for 6 students, and calculate the average score for each students, and calculate the average score for each course.

3) Add new functions based on your understanding of the student management system, such as adding a student's information, deleting a student's information, modifying a student's information, and searching for a student's information.

lab 8 file operations

lab purpose:

1. Master the basic concepts of files, such as the basic concepts of file input/output, the two organizational forms of files, the general steps of file operation, and so on.
2. Learn the basic algorithms for file operations, such as read/write, append, etc.
3. Understand the reading and writing methods of files, such as sequential reading and writing files or random reading and writing files

lab contents:

1. Enter a string from the keyboard, add the number 5 to the characters in the string, and save it to the file encryption.txt. Try writing a program (encryption program).
2. Read the characters in the file encryption.txt one by one, subtract the number 5, and display them on the screen. Try writing a program (decryption program).
3. Input the data of three students from the keyboard, store them in the file student.dat, and then read the data from the file and display it on the screen. Student data includes student ID, name, gender, and age. Try writing a program.