电子科技大学信息与软件工程学院

**实 验 报 告**

学 号

姓 名

（实验） 课程名称 C语言程序设计

理论教师 张学

实验教师 张学

**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名： 学号： 指导教师：张学**

**实验地点：科研楼A504 实验时间：16.11.22**

1. **实验名称：C语言的字符和数组处理技术**
2. **实验学时：3学时**
3. **实验目的：**

掌握C语言的字符处理技术。

掌握C语言的数组使用方法和技术。

掌握C语言函数的用法。

1. **实验原理：**

1.char类型的变量能被赋值为任何单个字符，字符常量需要用单引号括起来，而不是双引号。字符常量实际上是int类型，而不是char类型。

C语言允许使用单词signed和unsigned来修饰char类型：

signed char sch;

unsigned char uch;

转义序列共有两种：字符转义序列和数字转义序列。字符转义序列使用起来很方便，但是没有包含所有无法打印的ASCII字符；数字转义序列可以表示任何字符。

2.数组是含有多个数据值的数据结构，并且每个数据值具有相同的数据类型。这些数据值被称为元素。

一维数组中的元素一个接一个地编排在单独一行内。

为了声明一个数组，需要说明数组元素的类型和数量；数组的元素可以是任何类型；长度为n的数组，其元素的索引是从0到 n-1。

运算符sizeof可以确定数组的大小(字节数)；还可以用sizeof来计算数组元素的大小，比如a[0]。

3.函数简单来说就是一连串组合在一起并且命名的语句。每个形式参数需要说明其类型；形式参数间用逗号进行分隔；如果函数没有形式参数，那么在圆括号内应该出现void；函数体可以包含声明和语句。

4.实际参数：在C语言中，实际参数是通过值传递的：调用函数时，计算出每个实际参数的值并且把它赋值给相应的形式参数。函数执行过程中，对形式参数的改变不会影响实际参数的值。

1. **实验内容：**

项目1：第7章编程题4

项目2：第7章编程题12

项目3：第8章编程题2

项目4：第8章编程题9

项目5：第8章编程题15

项目6：第9章编程题3

1. **实验器材（设备、元器件）：**

个人电脑一台

1. **实验步骤：**

1、明确项目需求

2、编写代码

3、编译代码

4、测试程序

5、根据测试结果对程序进行调试改进

1. **实验结果与分析（含重要数据结果分析或核心代码流程分析）**

**项目1**

#include <stdio.h>

int main() {

char c;

printf("Enter phone number:");

while((c = getchar()) != '\n'){

switch(c){

case 'A': case 'B': case 'C':

printf("2");

break;

case 'D': case 'E': case 'F':

printf("3");

break;

case 'G': case 'H': case 'I':

printf("4");

break;

case 'J': case 'K': case 'L':

printf("5");

break;

case 'M': case 'N': case 'O': case 'Q':

printf("6");

break;

case 'P': case 'R': case 'S':

printf("7");

break;

case 'T': case 'U': case 'V':

printf("8");

break;

case 'W': case 'X': case 'Y': case 'Z':

printf("9");

break;

default: putchar(c);

break;

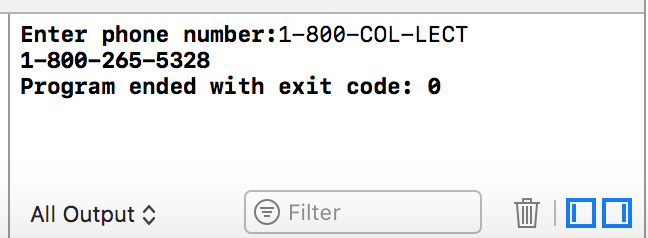
}

}

return 0;

}

测试结果：



实验结论：测试正确，程序满足项目需求。

case 'W': case 'X': case 'Y': case 'Z':

printf("9");

break;

default: putchar(c);

break;

}

}

return 0;

}

项目2

#include <stdio.h>

int main() {

int ch;

float a, b;

printf("Enter an expression:");

scanf("%f", &a);

while ((ch = getchar ()) != '\n'){

scanf ("%f", &b);

switch (ch){

case '+':

a += b;

break;

case '-':

a -= b;

break;

case '\*':

a \*= b;

break;

case '/':

a /= b;

break;

}

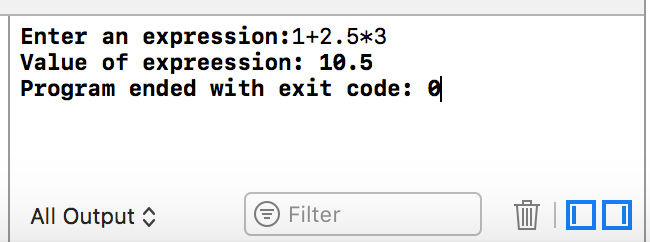
}

printf ("Value of expreession: %.1f\n", a);

return 0;

}

测试结果：



实验结论：测试正确，程序满足项目需求。

项目3

#include <stdio.h>

#include <stdbool.h>

int main() {

int digit\_seen[10] = {0};

int digit, i;

long n;

printf ("Enter a number: ");

scanf("%ld", &n);

while (n >0) {

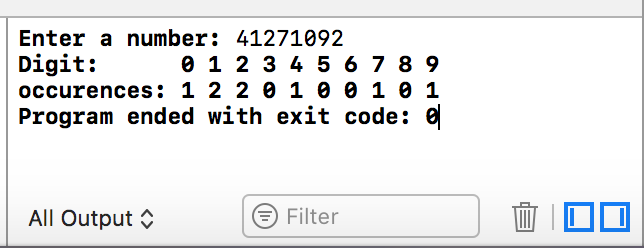
digit = n % 10;

digit\_seen[digit] += 1;

n /= 10;

}

测试结果：



实验结论：测试正确，程序满足项目需求。

printf ("Digit: 0 1 2 3 4 5 6 7 8 9\n");

printf ("occurences: ");

for (i = 0; i < 10; i++){

printf ("%d ", digit\_seen[i]);

}

printf ("\n");

return 0;

}

项目4

int main() {

int ch = 65, direction, m = 0, n = 0;

int i,j;

char a[N][N];

for(i=0;i<N;i++)

{

for(j=0;j<N;j++){

a[i][j] = '.';

}

}

a[0][0] = 'A';

srand((unsigned) time (NULL));

for (; ch < 90; ){

direction = rand () % 4;

switch (direction){

case 0: {if ((m-1) > 0 && a[m-1][n] == '.')

m--;

else

continue;

}

break;

case 1: {if ((m+1) < 10 && a[m+1][n] == '.')

m++;

else

continue;

}

break;

case 2: {if ((n-1) > 0 && a[m][n-1] == '.')

else

continue;

}

break;

case 2: {if ((n-1) > 0 && a[m][n-1] == '.')

n--;

else

continue;

}

break;

case 3: {if ((n+1) < 10 && a[m][n+1] == '.')

n++;

else

continue;

}

break;

}

ch++;

a[m][n] = ch;

}

for(i=0;i<N;i++)

{

for(j=0;j<N;j++){

printf("%c ",a[i][j]);

}

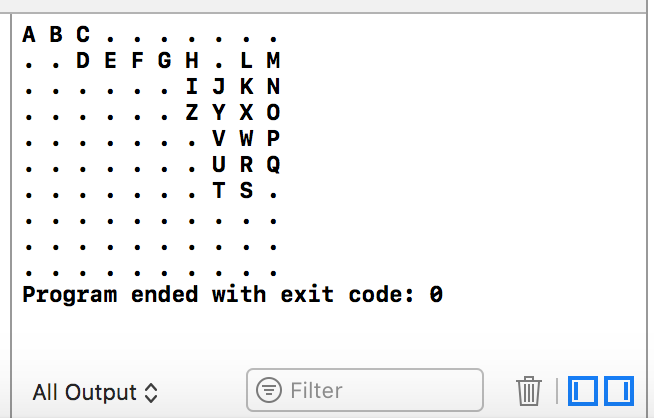
printf("\n");

}

return 0;

}

测试结果：



实验结论：测试正确，程序满足项目需求。

项目5

#include <stdio.h>

int main() {

int n, ch, i = 0;

char c[80];

printf ("Enter message to be encrypted:");

while ((ch = getchar()) != '\n'){

c[i] = ch;

i++;

}

c[i+1] = '\n';

printf ("Enter the shift amount (1-25):");

scanf ("%d", &n);

printf ("Encrypted message:");

for (i=0; ; i++){

if (c[i] != '\n'){

if (c[i] >= 65 && c[i] <= 90){

c[i] =((c[i] - 'A') + n) % 26 + 'A';

}

if (c[i] >= 97 && c[i] <= 122){

c[i] =((c[i] - 'a') + n) % 26 + 'a';

}

}

else {

break;

}

}

for (i=0; ; i++){

if (c[i] != '\n'){

printf ("%c", c[i]);

}

else {

break;

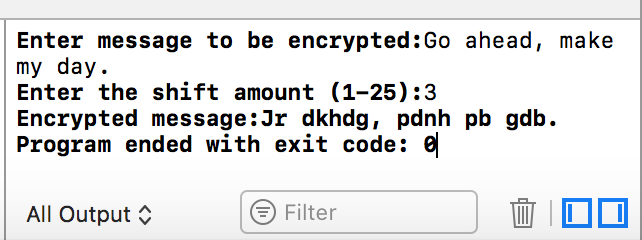
}

}

printf ("\n");

return 0;

测试结果：



实验结论：测试正确，程序满足项目需求。

printf ("\n");

return 0;

}

项目6

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void generate\_radom\_walk(char walk[10][10]);

void print\_array(char walk[10][10]);

void generate\_radom\_walk(char walk[10][10]){

int i, j;

int ch = 65, direction, m = 0, n = 0;

for(i=0;i<10;i++)

{

for(j=0;j<10;j++){

walk[i][j] = '.';

}

}

walk[0][0] = 'A';

srand((unsigned) time (NULL));

for (; ch < 90; ){

direction = rand () % 4;

switch (direction){

case 0: {if ((m-1) > 0 && walk[m-1][n] == '.')

m--;

else

continue;

}

break;

case 1: {if ((m+1) < 10 && walk[m+1][n] == '.')

m++;

else

continue;

}

break;

case 2: {if ((n-1) > 0 && walk[m][n-1] == '.')

n--;

else

continue;

}

break;

case 3: {if ((n+1) < 10 && walk[m][n+1] == '.')

n++;

else

continue;

}

break;

}

ch++;

walk[m][n] = ch;

}

return ;

}

void print\_array(char walk[10][10]){

int i, j;

for(i=0;i<10;i++)

{

for(j=0;j<10;j++){

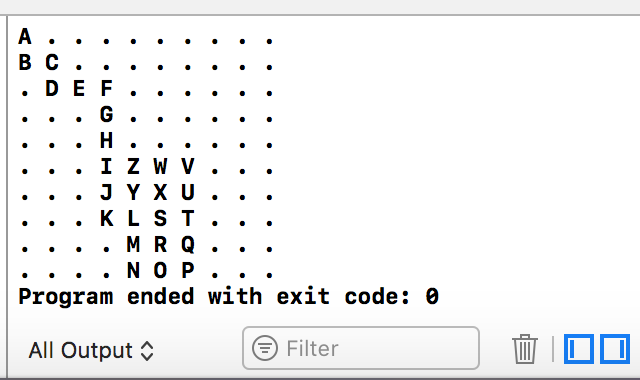
printf("%c ",walk[i][j]);

}

printf("\n");

}

测试结果：



实验结论：测试正确，程序满足项目需求。

return ;

}

int main() {

char a[10][10];

generate\_radom\_walk( a );

print\_array( a );

return 0;

}

1. **总结及心得体会：**

总结：

项目1考察的是对字符的理解和运用。用getchar函数读入和用switch语句判断是否为字母并将其改为相应的数字，其他字符不做改变的输出

项目2因为不知道要输入多少数，采用循环的形式。然而最后读入的是数字，循环里就采用字符，数字的形式。就需要先将第一个数字在循环外读入。再用switch语句判断符号做相应的运算。

项目3采用除10取余的方法取出每一个数字，然后将此数字作为数组下标访问数组元素，并将此元素加一计数，最后打印数组即可。

项目4先将数组初始化为点，然后将第一位改为A，再调用产生随机数的函数产生随机方向，再用switch语句判断。假如上移，则判断原位置是否在第一行并且上面的元素是否为点，若不符合条件，则继续产生方向。其他方向与之类似。最后再使用2重循环打印数组。

项目5采用字符数组存储输入的字符，最重要的在于要在用读入所有的字符后，还要加一个换行符，换行符的作用是在循环里终止循环。此题还需注意将字符移位的方法。

项目6是对项目4的改进，定义了打印数组和初始化和随机移位的函数。这两个函数的基本思想和项目4差不多，只是需要将函数中变量的名字稍作修改。

心得体会：这次的实验中遇到了不少问题。首先是知识掌握不牢固，导致在很多简单的地方浪费了很多的时间，经验教训就是应该先把知识掌握理解了再将其进行运用。还有就是随机数组打印字母的程序遇到了很多问题，直接拿到题目就在编译器里写，写着写着也就忘了本来的思路，所以以后应该先在草稿纸上写下主要的思路和重要复杂的步骤再上机编写程序；这个程序也反映出我思路不清晰，考虑问题不全面的问题，很难将他给的要求综合起来，综合起来后也是各种矛盾，还会发生越界的现象。

这次的实验锻炼了我的思路，也让我学会了应该怎样将问题考虑得更全面，这次的实验也有很多同学的帮助，大家不会的一起讨论交流就会启发我的灵感，所以团结合作有时还是必须的。最后，最重要的应该是我应该提升自己的能力，加强练习，变得更强。

1. **对本实验过程及方法、手段的改进建议：无**

**报告评分：**

**指导教师签字：**