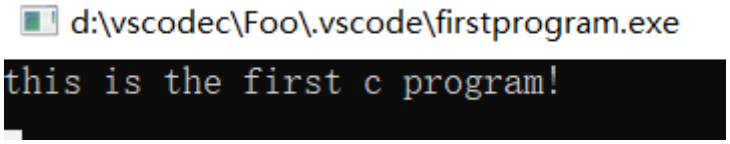
C语言从入门到精通

笔记本: <Inbox>
创建时间: 2020/7/6 10:51 **更新时间:** 2020/7/15 15:52
作者: 2494075190@qq.com

1第一个C语言程序

```
#include <stdio.h>
main()
{
    printf("this is the first c program!\n");
    getchar();
}
```

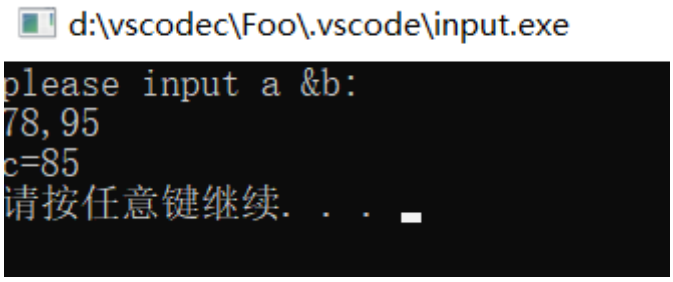


2输入一个整数，计算立方值

```
#include <stdio.h>
#include <stdlib.h>
main()
{
    int x, s;
    scanf("%d", &x);
    s = x * x * x;
    printf("three x add is=%d", s);
    system("pause");
}
```

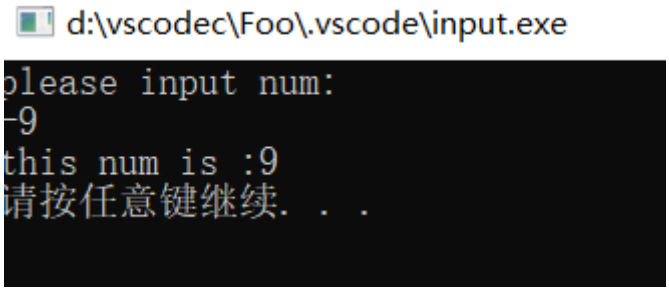
3两个整数，组成一个新的数

```
#include <stdio.h>
#include <stdlib.h>
int getResult(int a, int b)
{
    int m=0;
    m = a % 10 *10 + b % 10;
    return m;
}
main()
{
    int a=0, b=0, c=0;
    printf("please input a &b:\n");
    scanf("%d,%d", &a, &b);
    c = getResult(a,b);
    printf("c=%d\n", c);
    system("pause");
}
```



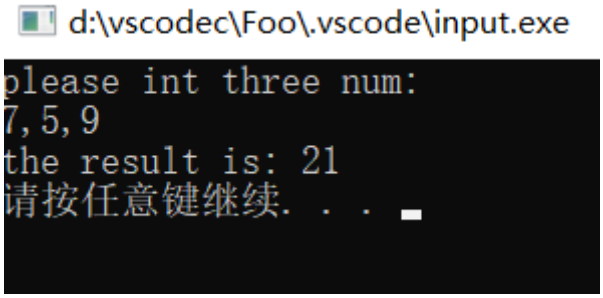
4求绝对值

```
#include <stdio.h>
#include <stdlib.h>
int getNum(int a)
{
    int m=0;
    if (a > 0)
    {
        m = a;
    }
    else
    {
        m = a * -1;
    }
    return m;
}
main()
{
    int a=0, result=0;
    printf("please input num: \n");
    scanf("%d", &a);
    result = getNum(a);
    printf("this num is :%d\n", result);
    system("pause");
}
```



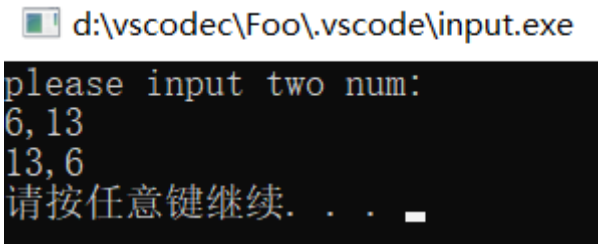
5计算三个数之和

```
#include <stdio.h>
#include <stdlib.h>
int getAdd(int a, int b, int c)
{
    int m;
    m = a + b + c;
    return m;
}
main()
{
    int x, y, z;
    int result;
    printf("please int three num:\n");
    scanf("%d,%d,%d", &x, &y, &z);
    result = getAdd(x, y, z);
    printf("the result is: %d\n", result);
    system("pause");
}
```



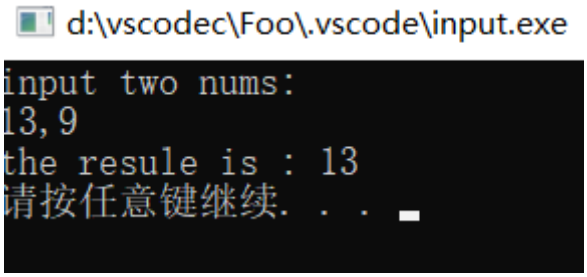
6两个值交换

```
#include <stdio.h>
#include <stdlib.h>
main()
{
    int a, b;
    int m;
    printf("please input two num:\n");
    scanf("%d,%d", &a, &b);
    m = a;
    a = b;
    b = m;
    printf("%d,%d\n", a, b);
    system("pause");
}
```



7比较两个值的大小

```
#include <stdio.h>
#include <stdlib.h>
int getMax(int a, int b)
{
    int m;
    if (a > b)
    {
        m = a;
    }
    else
    {
        m = b;
    }
    return m;
}
main()
{
    int x, y;
    int result;
    printf("input two nums:\n");
    scanf("%d,%d", &x, &y);
    result = getMax(x, y);
    printf("the resule is : %d\n", result);
    system("pause");
}
```



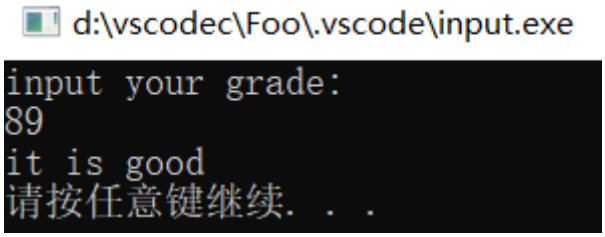
8常量算法

```
#include<stdio.h>
#include<stdlib.h>
#define MAX   10;
main(){
    int a,result;
    printf("input nums:\n");
    scanf("%d",&a);
    result = a * MAX;
    printf("the result:%d\n",result);
    system("pause");
```

```
}
```

9成绩排名

```
#include <stdio.h>
#include <stdlib.h>
main()
{
    int grade;
    int m;
    printf("input your grade:\n");
    scanf("%d", &grade);
    m = grade / 10;
    switch (m)
    {
    case 10:
    case 9:
        printf("good boy\n");
        break;
    case 8:
    case 7:
        printf("it is good\n");
        break;
    case 6:
        printf("it is ok\n");
        break;
    default:
        printf("you need study\n");
        break;
    }
    system("pause");
}
```



10成绩判断

```
#include <stdio.h>
#include <stdlib.h>
```

```
main()
{
    int grade;
    printf("input nums:\n");
    scanf("%d", &grade);
    if (grade < 60)
    {
        printf("it is bad\n");
    }
    else if (grade < 80)
    {
        printf("it is ok\n");
    }
    else if (grade < 100)
    {
        printf("good boy\n");
    }
    else
    {
        printf("error grade\n");
    }
    system("pause");
}
```

11while循环

```
#include <stdio.h>
#include <stdlib.h>
main()
{
    int a = 0;
    while (a < 10)
    {
        printf("a=%d\n", a);
        a++;
    }
    system("pause");
}
```

12do-while循环

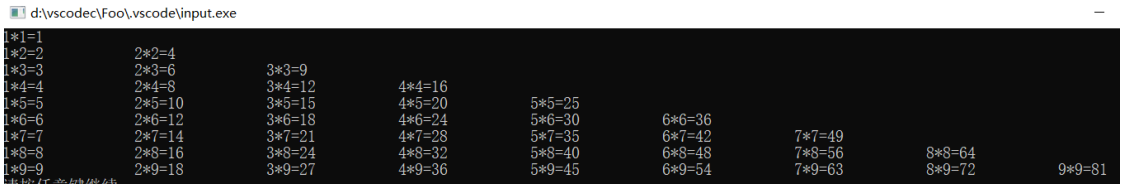
```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int b = 0;
    do
    {
        printf("b=%d\n", b);
        b++;
    } while (b < 10);
    system("pause");
    return 0;
}
```

13for循环

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    for (int c = 0; c < 10; c++)
    {
        printf("c=%d\n", c);
    }
    system("pause");
    return 0;
}
```

13九九乘法表

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    for (int i = 1; i < 10; i++)
    {
        for (int j = 1; j < i + 1; j++)
        {
            printf("%1d*%1d=%-4d", j, i, i * j);
            printf("\t");
        }
        printf("\n");
    }
    system("pause");
    return 0;
}
```



14计算0-10素数和素数之和

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int n, i;
    int s = 0;
    for (n = 3; n <= 10; n++)
    {
        for (i = 2; i < n; i++)
        {
            if (n % i == 0)
            {
                break;
            }
        }
        if (i >= n)
        {
            printf("%d\t", n);
            s = s + n;
        }
    }
    printf("\n");
    printf("s=%d\n", s);
    system("pause");
    return 0;
}
```

15计算最大公约数和最小公倍数

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int m, n, r, t;
    int x, y;
    printf("input m & n:\n");
    scanf("%d,%d", &m, &n);
    if (m < n)
    {
        t = m;
```

```
        m = n;
        n = t;
    }
    x = m;
    y = n;
    r = m % n;
    while (r > 0)
    {
        m = n;
        n = r;
        r = m % n;
    }
    printf("max = %d\n", n);
    printf("min = %d\n", x * y / n);
    system("pause");
    return 0;
}
```

16斐波那契数列

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int f1 = 1, f2 = 1, f3, i;
    printf("\t%d\t%d\t", f1, f2);
    for (i = 3; i <= 20; i++)
    {
        f3 = f2 + f1;
        printf("%d", f3);
        f1 = f2;
        f2 = f3;
        if (i % 5 == 0)
        {
            printf("\n");
        }
        printf("\t");
    }
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
1       1       2       3       5
8       13      21      34      55
89      144     233     377     610
987     1597    2584    4181    6765
请按任意键继续. . .
```

17猴子吃桃问题

```
#include<stdio.h>
#include<stdlib.h>
int main(int argc, char const *argv[])
{
    int i,k=1;
    int x;
    for(i=9;i>=1;i--){
        x =2*(k+1);
        printf("days:%d\t,nums:%d",i,x);
        k=x;
        printf("\n");
    }

    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
days:9  ,nums:4
days:8  ,nums:10
days:7  ,nums:22
days:6  ,nums:46
days:5  ,nums:94
days:4  ,nums:190
days:3  ,nums:382
days:2  ,nums:766
days:1  ,nums:1534
请按任意键继续. . .
```

17阶乘

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int i, j;
    int n;
    int s = 0;
    int t = 1;
    printf("input n:\n");
    scanf("%d", &n);
    for (i = 1; i <= n; i++)
    {
        t = 1;
```

```
        for (j = 1; j <= i; j++)
        {
            t = t * j;
        }
        s = s + t;
        printf("s=%d\n", s);
    }
    system("pause");
    return 0;
}
```

18数组声明和赋值

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int array[5];
    for (int i = 0; i < 5; i++)
    {
        array[i] = i * i;
        printf("%d\t", array[i]);
    }
    system("pause");
    return 0;
}
```

19 .数组初始化

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int array[5];
    int array2[] = {7, 8, 2, 19, 6};
    int m = 0;
    for (int i = 0; i < 5; i++)
    {
        array[i] = i * i;
        printf("%d\t", array[i]);
    }
    printf("\n");
    for (int j = 4; j >= 0; j--)
```

```
    {
        printf("%d\t", array[j]);
    }
    printf("\n");
    for (int k = 0; k < 5; k++)
    {
        printf("%d\t", array2[k]);
    }
    printf("\n");
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
0       1       4       9       16
16      9       4       1       0
7       8       2       19      6
请按任意键继续. . .
```

20数组算法

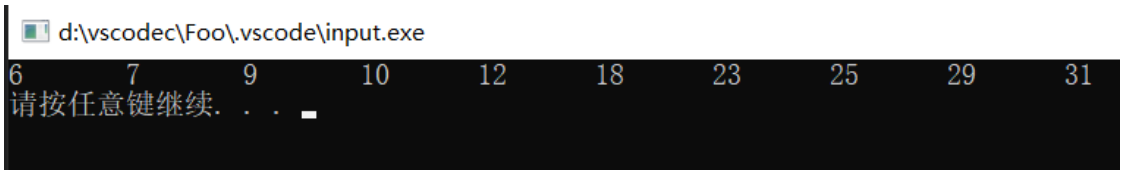
```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int n[100];
    int i, a1, a2, j = 0;
    for (i = 1; i <= 100; i++)
    {
        a1 = i % 10;
        a2 = i / 10;
        if (i % 3 == 0 && (a1 == 5 || a2 == 5))
        {
            n[j] = i;
            j++;
        }
    }
    printf("we need:%d\t", j);
    printf("\n");
    for (i = 0; i < j; i++)
    {
        printf("%d\t", n[i]);
    }
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
we need:6
15      45      51      54      57      75      请按任意键继续. . .
```

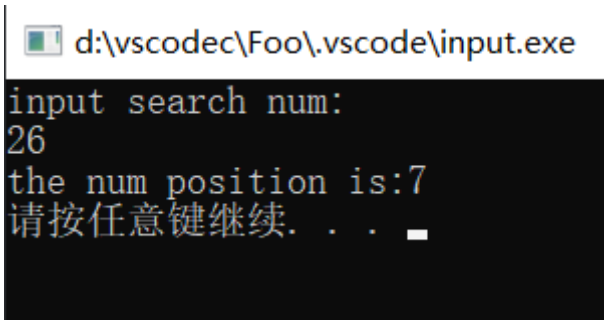
冒泡排序法

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a[10] = {12, 18, 9, 7, 6, 25, 23, 29, 31, 10};
    int i, j, temp;
    for (i = 0; i < 9; i++)
    {
        for (j = 0; j < 9 - i; j++)
        {
            if (a[j] > a[j + 1])
            {
                temp = a[j];
                a[j] = a[j + 1];
                a[j + 1] = temp;
            }
        }
    }
    for (i = 0; i < 10; i++)
    {
        printf("%d\t", a[i]);
    }
    system("pause");
    return 0;
}
```



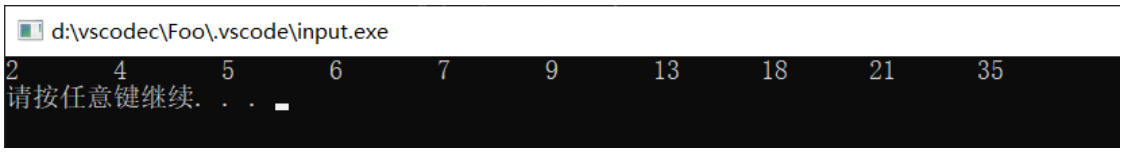
查找数组元素下标

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int x, i;
    int a[10] = {7, 9, 12, 16, 21, 5, 20, 26, 3, 38};
    printf("input search num:\n");
    scanf("%d", &x);
    for (i = 0; i < 10; i++)
    {
        if (x == a[i])
        {
            printf("the num position is:%d\n", i);
            break;
        }
    }
    system("pause");
    return 0;
}
```



选择排序法

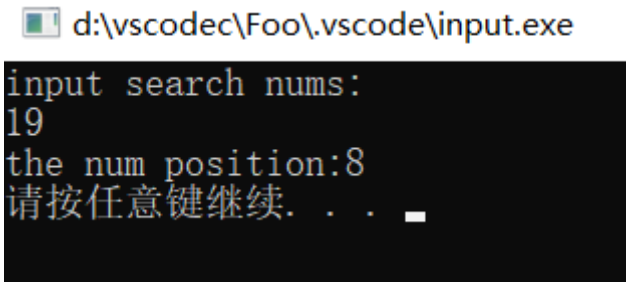
```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a[10] = {4, 9, 6, 7, 13, 5, 18, 21, 2, 35};
    int i, j, k, t;
    for (i = 0; i < 9; i++)
    {
        k = i;
        for (j = i + 1; j < 10; j++)
        {
            if (a[j] < a[k])
            {
                k = j;
            }
        }
        t = a[k];
        a[k] = a[i];
        a[i] = t;
    }
    for (i = 0; i < 10; i++)
    {
        printf("%d\t", a[i]);
    }
    system("pause");
    return 0;
}
```



二分法查找

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a[10] = {7, 12, 16, 9, 3, 24, 18, 5, 19, 36};
    int x;
    int bot, mid, top;
    printf("input search nums:\n");
    scanf("%d", &x);
    top = 0;
    bot = 9;
    mid = (top + bot) / 2;
    while (bot > top)
```

```
    {
        if (x == a[mid])
        {
            printf("the num position:%d\n", mid);
            break;
        }
        else
        {
            if (x > a[mid])
            {
                top = mid + 1;
                mid = (top + bot) / 2;
            }
            else
            {
                bot = mid - 1;
                mid = (bot + bot) / 2;
            }
        }
    }
    if (bot <= top)
    {
        printf("no found");
    }
    system("pause");
    return 0;
}
```



有序数组插入元素

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a[7] = {7, 13, 15, 19, 26, 33, 38};
    int x, i;
    printf("input your search num:\n");
    scanf("%d", &x);
    for (i = 5; i >= 0; i--)
    {
        if (a[i] > x)
        {
            a[i + 1] = a[i];
        }
        else
        {
            break;
        }
    }
    a[i + 1] = x;
    for (i = 0; i < 7; i++)
    {
        printf("%d\t", a[i]);
    }
     printf("\n");
    system("pause");
```

```
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
input your search num:
16
7       13      15      16      19      26      33
请按任意键继续. . .
```

删除有序数组元素

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a[7] = {3, 5, 9, 13, 16, 17, 26};
    int x, i;
    printf("please input delete num:\n");
    scanf("%d", &x);
    for (i = 0; i < 7; i++)
    {
        if (a[i] == x)
        {
            break;
        }
    }
    for (; i < 8 - 1; i++)
    {
        a[i] = a[i + 1];
    }
    printf("after delete:\n");
    for (i = 0; i < 6; i++)
    {
        printf("%d\t", a[i]);
    }
    printf("\n");
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
please input delete num:
13
after delete:
3       5       9       16      17      26
请按任意键继续. . .
```

字符数组声明

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
```

```
{
    char a[10] = {"hello"};
    for (int i = 0; i < 10; i++)
    {
        printf("%c", a[i]);
    }
    printf("\n");
    system("pause");
    return 0;
}
```

大小写转换

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"hello"};
    for (int i = 0; i < 10; i++)
    {
        if (a[i] > 0)
        {
            a[i] = a[i] - 32;
        }
    }
    for (int j = 0; j < 10; j++)
    {
        printf("%c", a[j]);
    }
    printf("\n");
    system("pause");
    return 0;
}
```

字符串连接

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"hello"};
```

```
    char b[10] = {"world"};
    printf("%s\n", strcat(a, b));
    system("pause");
    return 0;
}
```

字符串指定字符数连接

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"hello"};
    char b[10] = {"world"};
   // printf("%s\n", strcat(a, b));
    printf("%s\n",strncat(a,b,3));  // 三个参数  字符数组  字符串 n
    system("pause");
    return 0;
}
```

字符串复制

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"hello"};
    char b[10] = {"world"};
    strcpy(a, b); //把第二个参数，复制给第一个参数
    printf("%s\n", a);
    printf("%s\n", b);
    system("pause");
    return 0;
}
```

字符串部分复制

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"abcde"};
    char b[10] = {"vwxyz"};
    strncpy(a,b,3); //第三个参数表示字符串数目，第二个参数的字符串数目复制和替换给第一个参数
    printf("%s\n", a);
    printf("%s\n", b);
    system("pause");
    return 0;
}
```

字符串相等

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"abcdes"};
    char b[10] = {"abcdes"};
    int c = strcmp(a, b); //0表示相等
    printf("%d\n", c);
    system("pause");
    return 0;
}
```

字符串长度

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"abcdeswef"};
    char b[10] = {"abcdes"};
    int c = strlen(a);
    printf("%d\n", c);
    system("pause");
    return 0;
}
```

字符数组在内存中的字节数

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"abcdeswef"};
    char b[10] = {"abcdes"};
    int c = sizeof(a); //字符数组在内存中占用的字节数
    printf("%d\n", c);
    system("pause");
    return 0;
}
```

大写字母转换为小写字母

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"abcdeswef"};
    char b[10] = {"SNAKE"};
    printf("%s\n", strlwr(b));
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
snake
请按任意键继续. . .
```

大写转小写

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char a[10] = {"rabbit"};
    char b[10] = {"SNAKE"};
    printf("%s\n", strlwr(b));
    printf("%s\n", strupr(a));
    system("pause");
    return 0;
}
```

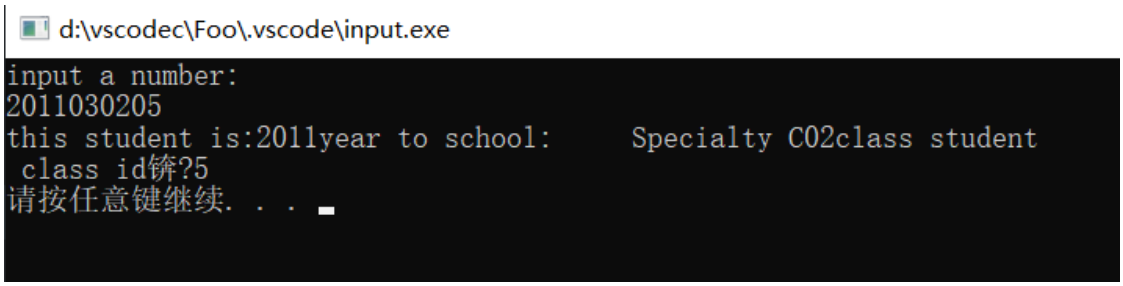
d:\vscodec\Foo\.vscode\input.exe

```
snake
RABBIT
请按任意键继续. . .
```

判断入学年份，字符拼接

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char sno[20];
    char syear[5];
    char sprof[3];
    char sclass[3];
    char snumber[3];
    char str[100] = "";
    int i = 0, l, flag = 0;
    printf("input a number:\n");
    scanf("%s", sno);
    l = strlen(sno);
    strncpy(syear, sno, 4);
    syear[4] = '\0';
    strcat(str, "this student is:");
    strcat(str, syear);
    strcat(str, "year to school:\t");
    for (i = 4; i <= 5; i++)
    {
        sprof[i - 4] = sno[i];
    }
    sprof[i - 4] = 0;
```

```
    for (i = 6; i <= 7; i++)
    {
        sclass[i - 6] = sno[i];
    }
    sclass[i - 6] = 0;
    for (i = 8; i < l; i++)
    {
        snumber[i - 8] = sno[i];
    }
    snumber[i - 8] = 0;
    i = 0;
    while (sprof[i] != '\0')
    {
        flag = flag * 10 + sprof[i] - 48;
        i++;
    }
    switch (flag)
    {
    case 1:
        strcat(str, "Specialty A");
        break;
    case 2:
        strcat(str, "Specialty B");
        break;
    case 3:
        strcat(str, "Specialty C");
        break;
    case 4:
        strcat(str, "Specialty D");
        break;
    case 5:
        strcat(str, "Specialty E");
        break;
    case 6:
        strcat(str, "Specialty F");
        break;
    case 7:
        strcat(str, "Specialty G");
        break;
    case 8:
        strcat(str, "SpecialtyH");
        break;
    case 9:
        strcat(str, "Specialty I");
        break;
    case 10:
        strcat(str, "Specialty J");
        break;
    default:
        break;
    }
    strcat(str,sclass);
    strcat(str,"class student");
    strcat(str,"\n class id:");
    strcat(str,snumber);
    puts(str);
    system("pause");
    return 0;
}
```



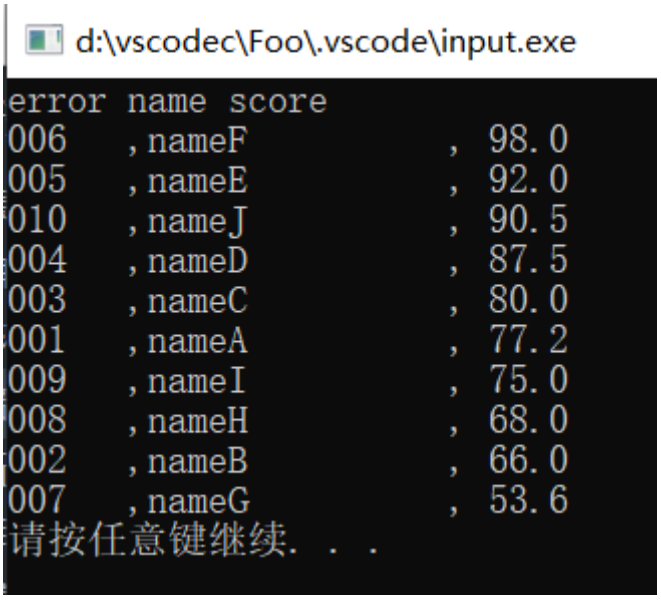
书名按首字母排序

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
{
    char book[5][20] = {{"javascript"}, {"android"}, {"ruby"}, {"kotlin"},
{"sql"}};
    char t[20];
    int i, j;
    for (i = 0; i < 4; i++)
    {
        for (j = 0; j < 5 - i - 1; j++)
        {
            if (strcmp(book[j], book[j + 1]) > 0)
            {
                strcpy(t, book[j]);
                strcpy(book[j], book[j + 1]);
                strcpy(book[j + 1], t);
            }
        }
        for (i = 0; i < 5; i++)
        {
            puts(book[i]);
        }
    }
    system("pause");
    return 0;
}
```

成绩排名：

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(int argc, char const *argv[])
```

```
{
    char sname[10][15] = {"nameA", "nameB", "nameC", "nameD", "nameE", "nameF",
"nameG", "nameH", "nameI", "nameJ"};
    char sno[10][5] = {"001", "002", "003", "004", "005", "006", "007", "008",
"009", "010"};
    float sco[10] = {77.2, 66, 80, 87.5, 92, 98, 53.6, 68, 75, 90.5};
    char t[10];
    float x;
    int i, j;
    for (i = 0; i < 9; i++)
    {
        for (j = 0; j < 10 - 1 - i; j++)
        {
            if (sco[j] < sco[j + 1])
            {
                x = sco[j];
                sco[j] = sco[j + 1];
                sco[j + 1] = x;
                strcpy(t, sno[j]);
                strcpy(sno[j], sno[j + 1]);
                strcpy(sno[j + 1], t);
                strcpy(t, sname[j]);
                strcpy(sname[j], sname[j + 1]);
                strcpy(sname[j + 1], t);
            }
        }
    }
    printf("error name score\n");
    for (i = 0; i < 10; i++)
    {
        printf("%-6s,%-15s,%5.1f\n", sno[i], sname[i], sco[i]);
    }
    system("pause");
    return 0;
}
```



函数

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
```

```
    int m, n, t;
    int i;
    long facm = 1, facn = 1, facmn = 1, c;
    printf("please input m and n:\n");
    scanf("%d,%d", &m, &n);
    if (m < n)
    {
        t = m;
        m = n;
        n = t;
    }
    for (i = 1; i <= m; i++)
    {
        facm = facm * i;
    }
    for (i = 1; i <= n; i++)
    {
        facn = facn * i;
    }
    for (i = 1; i < m - n; i++)
    {
        facmn = facmn * i;
    }
    c = facm / (facn * facmn);
    printf("combonation is %ld \n", c);
    system("pause");
    return 0;
}
```

判断字符大小写

```
#include <stdio.h>
#include <stdlib.h>
int judge(char c)
{
    if (c >= 'A' && c <= 'Z')
    {
        return 1;
    }
    else if (c >= 'a' && c <= 'z')
    {
        return 2;
    }
    else
    {
        return 0;
    }
}
int main(int argc, char const *argv[])
{
    char ch;
    printf("input a character:\n");
    scanf("%c", &ch);
    printf("%d\n", judge(ch));
    printf("\n");
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
input a character:
S
1

请按任意键继续. . .
```

求特定范围内的素数之和

```
#include <stdio.h>
#include <stdlib.h>
int prime(int m)
{
    int i;
    for (i = 2; i < m; i++)
    {
        if (m % i == 0)
        {
            return 0;
        }
    }
    if (i >= m)
    {
        printf("%d", m);
        return 1;
    }
}
int main(int argc, char const *argv[])
{
    int x, y, i, t;
    int s = 0;
    printf("input x &y:\n");
    scanf("%d,%d", &x, &y);
    if (x > y)
    {
        t = x;
        x = y;
        y = t;
    }
    for (i = x; i <= y; i++)
    {
        if (prime(i))
        {
            s += i;
        }
    }
    printf("\n");
    printf("the sum of prime number in x and y is :%d\n", s);
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

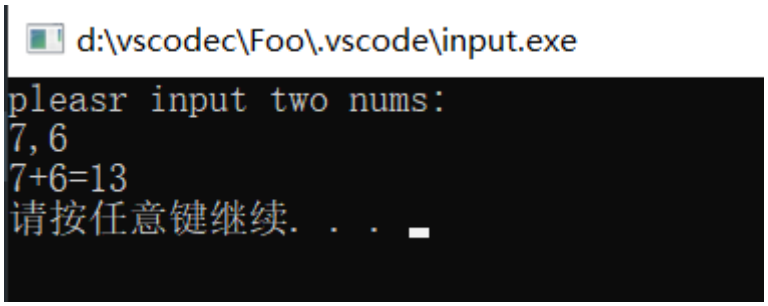
```
input x &y:
3,12
3       5       7       11
the sum of prime number in x and y is :26
请按任意键继续. . .
```

小写字母转换为大写字母

```
#include<stdio.h>
#include<stdlib.h>
char change(char c){
    c=c-32;
    return c;
}
int main(int argc, char const *argv[])
{
    char ch;
    printf("input a character:\n");
    ch = getchar();
    if (ch >= 'a' && ch <= 'z')
    {
        ch = change(ch);
    }
    printf("%c\n",ch);
    system("pause");
    return 0;
}
```

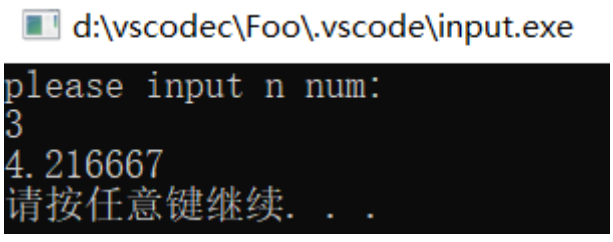
两个数求和

```
#include <stdio.h>
#include <stdlib.h>
int add(int x, int y)
{
    int m;
    m = x + y;  //形参
    return m;
}
int main(int argc, char const *argv[])
{
    int a, b, c;
    printf("pleasr input two nums:\n");
    scanf("%d,%d", &a, &b);
    c = add(a, b);   //实参
    printf("%d+%d=%d", a, b, c);
    printf("\n");
    system("pause");
    return 0;
}
```



编写函数计算

```
#include <stdio.h>
#include <stdlib.h>
float getFn(int x)
{
    float m;
    if (x < 0)
    {
        m = (float)(x - 1) / (x - 2);
    }
    else if (x == 0 || x == 2)
    {
        m = 0;
    }
    else
    {
        m = (float)(x + 1) / (x - 2);
    }
    return m;
}
int main(int argc, char const *argv[])
{
    int n;
    float s;
    printf("please input n num:\n");
    scanf("%d", &n);
    for (int i = -n; i < n + 1; i++)
    {
        s += getFn(i);
    }
    printf("%f\t", s);
    printf("\n");
    system("pause");
    return 0;
}
```



递归计算阶乘

```
#include <stdio.h>
#include <stdlib.h>
int getResult(int n)
{
```

```
    int m;
    if (n > 1)
    {
        m = n * getResult(n - 1);
    }
    else
    {
        m = 1;
    }
    return m;
}
int main(int argc, char const *argv[])
{
    int n;
    long y;
    printf("input number:\n");
    scanf("%d", &n);
    y = getResult(n);
    printf("%d\t,%d\t", n, y);
    system("pause");
    return 0;
}
```

斐波拉契数列 递归

```
#include <stdio.h>
#include <stdlib.h>
int getNum(int n)
{
    // 1,1,2,3,5,8,13,21... 斐波拉契数列
    int s;
    if (n > 2)
    {
        s = getNum(n - 1) + getNum(n - 2);
    }
    else if (n == 1 || n == 2)
    {
        s = 1;
    }
    return s;
}
int main(int argc, char const *argv[])
{
    int x;
    int result;
    printf("input num:\n");
    scanf("%d", &x);
    result = getNum(x);
    printf("%d\t", result);
    printf("\n");
    system("pause");
    return 0;
}
```

全局变量

```
#include <stdio.h>
#include <stdlib.h>
int n1, n2, n3, n4;
void count(char s[])
{
    int i;
    for (i = 0; s[i] != 0; i++)
    {
        if (s[i] >= 'a' && s[i] <= 'z')
        {
            n1++;
        }
        else if (s[i] >= 'A' && s[i] <= 'Z')
        {
            n2++;
        }
        else if (s[i] >= '0' && s[i] <= '9')
        {
            n3++;
        }
        else
        {
            n4++;
        }
    }
}
int main(int argc, char const *argv[])
{
    char str[80];
    gets(str);
    count(str);
    printf("Low:%d\n", n1);
    printf("Big:%d\n", n2);
    printf("Num:%d\n", n3);
    printf("Other:%d\n", n4);
    system("pause");
    return 0;
}
```

条件编译

```
#include "stdio.h"
#include "stdlib.h"
#define FLAG 1
int main(int argc, char const *argv[])
{
    char s[80];
    int i;
    gets(s);
    #if FLAG
        for (i = 0; s[i] != 0; i++)
        {
            if (s[i] > 'a' && s[i] < 'z')
            {
                s[i] = s[i] - 32;
            }
        }
    #else
        for (i = 0; s[i] != 0; i++)
        {
            if (s[i] > 'A' && s[i] < 'Z')
            {
                s[i] = s[i] + 32;
            }
        }
    #endif
    puts(s);
    system("pause");
    return 0;
}
```

指针

```
#include "stdio.h"
#include "stdlib.h"
int main(int argc, char const *argv[])
{
    int a = 10;
    int *p;
    p = &a;
    printf("a is :%d\n", a);
    printf("p is : %d\n", *p);
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
a is :10
p is : 10
请按任意键继续. . .
```

指针赋值运算

```
#include "stdio.h"
#include "stdlib.h"
int main(int argc, char const *argv[])
{
    int a, *pa;
    float b, *pb;
    char c, *pc;
    float d, *pd;
    pa = &a;
    pb = &b;
    pc = &c;
    pd = &d;
    scanf("%d,%f,%c,%f", pa, pb, pc, pd);
    printf("%d\t%f\t%c\t%f\t", *pa, *pb, *pc, *pd);
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
23,5.68,G,63.2
23      5.680000        G       63.200001       请按任意键继续. . .
```

指针 数据交换

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a, b;
    int *pa = &a, *pb = &b;
    int t;
    scanf("%d,%d", pa, pb);
    t = *pa;
    *pa = *pb;
    *pb = t;
    printf("%d,%d", a, b);
    printf("\n");
    system("pause");
    return 0;
}
```

d:\vscodec\Foo\.vscode\input.exe

```
9,7
7,9
请按任意键继续. . .
```

数组赋值

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a[5] = {7, 6, 12, 9, 3};
    int *pa;
    pa = a;
    for (int i = 0; i < 5; i++)
    {
        printf("%d\n", pa[i]);
    }
    system("pause");
    return 0;
}
```

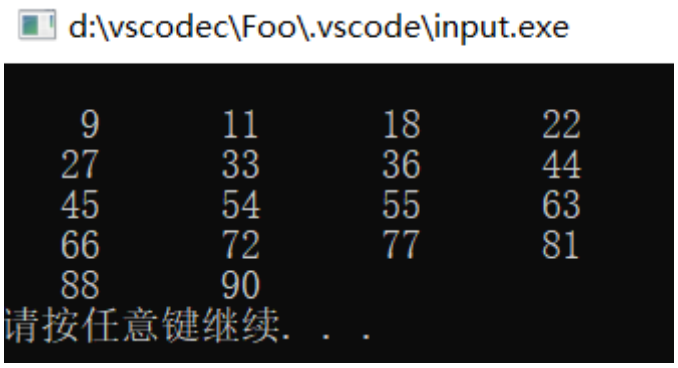
第二种写法，让指针指向下一个元素

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a[5] = {7, 6, 12, 9, 3};
    int *pa;
    pa = a;
    for (int i = 0; i < 5; i++)
    {
        printf("%d\n", *pa);
        pa++;
    }
    system("pause");
    return 0;
}
```

能被9整除 或者能被11整除

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a[1000], i, n = 0;
    int *p = a;
    for (i = 1; i <= 100; i++)
    {
        if ((i % 9 == 0 && i % 11 != 0) || (i % 11 == 0 && i % 9 != 0))
        {
```

```
            *p++ = i;
            n++;
        }
    }
    p = a;
    for (i = 0; i < n; i++, p++)
    {
        if (i % 4 == 0)
        {
            printf("\n");
        }
        printf("%5d\t", *p);
    }
    printf("\n");
    system("pause");
    return 0;
}
```



二维数组

```
#include <stdio.h>
#include <stdlib.h>
#define N 15
void fun(int *a, int *b, int *x)
{
    int i, j = 0;
    b[j] = a[0];
    for (i = 0; i < N; i++)
    {
        if (b[j] != a[i])
        {
            j++;
            b[j] = a[i];
        }
    }
    *x = j + 1;
}
int main(int argc, char const *argv[])
{
    int a[N] = {1, 1, 2, 2, 2, 3, 4, 4, 4, 7, 7, 7, 9, 11, 15};
    int b[15], n, i;
    fun(a, b, &n);
    for (i = 0; i < n; i++)
    {
        printf("%d\t", b[i]);
    }
    printf("\n");
    system("pause");
    return 0;
}
```

```
d:\vscodec\Foo\.vscode\input.exe
1       2       3       4       7       9       11      15
请按任意键继续. . .
```

结构体

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
struct s_score
{
    int no;
    char name[10];
    int score[3];
};
void output(struct s_score a);
void fun(int score[]);
void output(struct s_score a)
{
    int i;
    printf("%d,%s", a.no, a.name);
    for (i = 0; i < 3; i++)
    {
        printf("%d\t", a.score[i]);
    }
    printf("\n");
}
void fun(int score[])
{
    int i;
    for (i = 0; i < 3; i++)
    {
        score[i] = score[i] + 10;
        if (score[i] > 100)
        {
            score[i] = 100;
        }
    }
}
int main(int argc, char const *argv[])
{
    struct s_score a = {1001, "zhangsna", {6, 9, 2}};
    a.no = 1001;
    strcpy(a.name, "tony");


    a.score[0] = 17;
    a.score[1] = 13;
    a.score[2] = 19;
    output(a);
    fun(a.score);
    output(a);
    system("pause");
    return 0;
}
```

与运算符& 二进制两个同时都为1的结果为1

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a = 7, b = 13;
    int c;
    c = a & b;
    printf("%d\n", c);
    system("pause");
    return 0;
}
```

运行结果：5

或运算符 | 二进制，其中有一个为1的结果为1

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a = 7, b = 13;
    int c;
    c = a | b;
    printf("%d\n", c);
    system("pause");
    return 0;
}
```

运行结果：15

异或运算符 相同为0，不同为1

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char const *argv[])
{
    int a = 7, b = 13;
    int c;
    c = a ^ b;
    printf("%d\n", c);
    system("pause");
    return 0;
}
```

运行结果：10

ip地址由32位二进制组成，为了书写和描述方便，通常用十进制表示

十进制写法：192.168.1.1

二进制写法：1100　0000　1010　1000　0000 0001 0000 0001

255的二进制；1111　1111