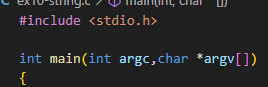
#### 初学预览：<https://open.alipay.com/portal/forum/post/138201017>

#### 程序结构

头文件

主函数



cmd 当前目录> ex10.exe 11 22 33

执行程序时，可以从命令行传递值给C程序，这些值称为**命令行参数**（可从外部控制程序，eg:选择功能），给main函数的列表赋值，int argc是传入参数的个数，char \*argv[]是存放参数的数组

数组的0下标位置是可执行文件名称，即argc[0]==ex10.exe

#### 语法**：结束语句用大括号 } 或分号 ;**

#### printf格式化输出，变量类型

//    printf("123","456")只输出第一对双引号的值，后面的双引号只能作为字符串数组输出

%d int 输出整数和十进制

%f float 整数部分+小数部分的有效数字是7位 \double

%c char单字符 用单引号包围 ‘a’

%s char[] aStringList[]字符串用数组表示 用双引号包围

%ld long 长整型 sizeof()返回值

打印空格’\0’

%o输出八进制

%x输出十六进制

%u unsigned 纯二进制，无符号整数，非负数，常用于二进制的位运算

Make 将.c文件编译成可执行文件

sizeof(int) 返回占用字节大小，返回值用%lu或%ld输出

变量的隐式转换和显示转换？？？

一个整型变量占4个字节

一个字符占一个字节

定义一个列表 type name[ ]={1,2,3}

#### scanf输入

scanf("must %s", &num);

输入变量时，不能忽略双引号内的字符(内部定义的输入格式)，eg:must 10

输入多个变量scanf("%d %s", &num,&str);

输入double类型，用%lf

#### 变量的命名规范

· 只能由字母（包括大写和小写）、数字和下划线（\_）组成。  
· 不能以数字开头  
· 长度不能超过63个字符  
· 变量名中区分大小写  
· 变量名不能使用关键字

#### 变量定义

变量类型 变量名=初始值

整数类型的内存占用范围：

char :1字节 -128~127

unsigned 类型表示整数不以补码的形式

例如 char a=255 输出-1 unsigned char a=255 输出255

short: 2字节 -32768~32767

Int: 取决于编译器（cpu），

long: 4字节

longlong: 8字节

c语言和python的区别

|  |  |
| --- | --- |
| **C** | **Pthon** |
| char str1='a';//单字符单引号 %c | 动态类型语言，无需声明变量类型  str1="单双引号都可" |
| char \*strlist="hello world"//字符串双引号，一维数组 %s | strlist=['aaa','bbbb','ccc'] |
| char \*states[]={  //二维数组 %s  "aaa","bbb","ccc","ddd"  }; | //二维数组  strllist=[[1,2,3],['a','b','c'],[1,1]] |
| Int argv[]={1,2,3,4};  //列表用花括号 | //列表用方括号 |

字符串=[字符数组](https://so.csdn.net/so/search?q=%E5%AD%97%E7%AC%A6%E6%95%B0%E7%BB%84&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/chenkaibsw/article/details/_blank)

char c[12] = {'I','a','m','a','b','o','y','.'};方括号里规定长度

char str[] = {'I','a','m','h','a','p','p','y','\0'};数组长度随机

char str[] = "I am happy"; 没有花括号的数组，str[0]='I'

char str[] = {"I am happy"};这个花括号可有可无

常用定义字符串数组方式 \*str=”aaa”

初始化一个字符串

Char str[ ]={ };

给字符数组赋值,为

Str[0]=”a”

strlen()函数求出的字符串长度为**有效长度**，既不包含字符串末尾结束符 ‘\0’；

sizeof()操作符求出的长度**包含**字符串末尾的结束符 ‘\0’；

#### 常量定义

const int ARGUMENT=10; 常量被定义后不可被修改

const 变量类型 全大写变量名=初始值

#### 运算符

% 取余，用%d输出

printf("%d",18/7);

结果为整数，需以%d输出

 printf("%f",18/7.0);

结果为浮点数，以%f输出

整数除小数，结果为小数

优先级(运算符计算的优先顺序)：

|  |  |
| --- | --- |
| 1 | 单目运算符（自加++自减--） |
| 2 | / \* % |
| 3 | + - |
| 4 | >= <= == != （关系运算符） |
| 5 | & ^ | && || |
| 6 | = (赋值运算符) |

**eg:** a\*-b=a\*负b 自右向左算

变量值交换（两个杯子中的水互换，需要借助第三个杯子）

a=5; b=6; c=a; a=b; b=c;

复合赋值

a+=b

递增递减

a++结果为a初始值, a等于a+1（a自加，输出a原值）

++a结果为a+1，a也等于a+1（a自加，输出a的自加值）

连续相同运算符，从左到右计算

**eg :** a==b==6 判断1与6是否相等或0与6是否相等

#### for循环

用于有固定次数的循环

|  |  |
| --- | --- |
| c语言 | Python |
| int i=0;  for(i=0;i<argc;i++){  //初始赋值，判断，累加          printf("arg %d:%s\n",i,argv[i]);      } | for i in range(10)://可迭代序列          print(i)  l1=[11,22,33]  for i in l1: //直接打印数组元素      print(i) |

#### While 循环

控制条件，执行循环体，用于无法确定执行次数的循环

|  |  |
| --- | --- |
| c | p |
| int i=10;  while(i<11){  printf("%d",i);  i=i+1;  } | i = 1   while i <= 5:       print(i)       i += 1 |

注：c语言同作用域定义同名变量会使编译报错,循环声明同名变量，

**do-while**

用于必须执行一次才能往下运行的循环

Eg:判断一个数是几位数，循环/10，累计

|  |  |
| --- | --- |
| **c** | **p** |
| int x=0;   int num;   scanf("%d",&num);  do{      x++;      num/=10;  }while(num);  printf("%d",x); | **无do-while循环**  一般在while用if判断，不符合循环条件  则break |

#### If语句

用级联的If-else-if实现多分支

|  |  |
| --- | --- |
| c | p |
| if(argc==1){       printf("");  }else if(argc>1 && argc<4){       printf("");  }else{      printf("");      } | x = 10  if x > 0:      print("x是正数")  elif x == 0:      print("x是零")  else:      print("x是负数") |

#### switch语句-多分支

执行与控制表达式结果相同的case

|  |  |
| --- | --- |
| **c** | **p** |
| int type;  scanf("%d",&type);  switch (type){ //type必须为整型          case 1:          case 2:              printf("2");              break;          case 3:              printf("2");              break;          default:              printf("大于1，2");              break;      } | **无switch语句** |

#### 字符编码

Unicode 万国码，统一码，给每个字符唯一的编码

UTF-8，变长编码，根据符号的不同，规定不同的存储长度

GB2312，中国发布，用于汉字处理

GBK，CB2312的扩展

GB18030，包含少数民族文字，变长多字节字符

Unicode⊆UTF-8

GB2312⊆GBK⊆GB18030

[字符编码的代码 命令行chcp查看](https://blog.csdn.net/a320976399/article/details/119793761)

#### 输出乱码原因及解决

1. 在命令行中运行，输出中英字符，数字，均为乱码，可将vsCode的编码改为GB2312解决
2. 在vsCode终端，用GB2312,不输出中文

用vsCode终端更方便



#### 函数

main函数为程序入口，不在main函数中定义或使用（main外部定义），在执行过程中有被调用的函数，需要预先做前向声明才可使用

// 前向声明，不在main函数定义但执行过程被调用，需预先声明要在程序中用的函数，在未定义前

int can\_print\_it(char ch);

void print\_letters(char arg[]);

#### 内存存储

以16进制或8进制作为内存地址，方便缩短长数字，进制数越大，数表达的长度越短

64，16，8为2的n次方，方便跟二进制进行转换

变量的本质是内存地址的符号化，获取内存地址的方式

#### 二进制

表示数字：最高位0为正数，1为负数

负数为整数的补码，也就是反码（1变0）后再加1，数字本身带符号参与运算，只用加法的方式运算，方便电路设计，带负号的1字节数字范围是 -128~0~127

二进制加法（逢二进一）：0111+0101=1100

#### 十六进制转十进制，余数定理

十六进制转十进制，权值计算，A(10),B(11),C(12),D(13),E(14),F(15)

2AF5=16^3x2+16^2x10+16^1x15+16^0x5

Cpu，寄存器和内存的关系：cpu从内存取出数据时首先放在寄存器中暂存（缓存），寄存器的大小与int的内存大小相等

、

RAM

CPU

总线

寄存器

#### 隐式类型转换（自动类型转换）

不同数据类型进行运算，会自动转换为相同的类型，且自下而上，转换为较大的数据类型

* char(1B) --> short(2B) --> int(4B=32bit) --> long(4B) --> long long(8B)

（32位系统的long内存占4B，64位占8B）

* int --> float(4B) --> double(8B)
* unsigned ----- %u 无符号整数，非负数，常用于二进制的位运算
* short ----- %hd
* long ------ %ld

**强制类型转换--**优先级高于运算符

(int)12.0

double result= (double) 5 / 2

随机数

#include <stdio.h>

#include <stdlib.h> //用于使用rand()

#include <time.h> //用于使用time()，获取当前系统时间，用于设置随机数种子以增加随机性

//主函数

int main(){

    srand(time(0)); //用于生成随机数种子

    int a = rand(); //rand()函数用于生成伪随机数

    printf("%d\n",a%10);  //取10以内的随机数

    return 0;

}