# 目录

[目录 1](#_Toc32179233)

[第1章 C语言基础 2](#_Toc32179234)

[1.1 C语言概述 2](#_Toc32179235)

[1.2 搭建开发环境 2](#_Toc32179236)

[1.2.1 GCC编译器 1073766401](#_Toc32179237)

[1.2.2 GCC编译器安装 1073766401](#_Toc32179238)

[1.2.3 GCC编译器验证是否安装 1073766401](#_Toc32179239)

[1.3 CMD使用方法 1073766401](#_Toc32179240)

[1.3.1 切换盘符 1073766401](#_Toc32179241)

[1.3.2 进入文件夹命令 1073766401](#_Toc32179242)

[1.3.3 查看文件夹名或文件名 1073766401](#_Toc32179243)

[1.3.4 清除命令行显示的信息 1073766401](#_Toc32179244)

[1.3.5 键盘上下键功能 1073766401](#_Toc32179245)

[1.3.6 gcc编译命令 1073766401](#_Toc32179246)

[1.4 C语言的程序结构 1073766401](#_Toc32179247)

[1.4.1 头文件包含 1073766401](#_Toc32179248)

[1.4.2 注释 1073766401](#_Toc32179249)

[1.4.3 函数注释头 1073766401](#_Toc32179250)

[1.4.4 主函数 1073766401](#_Toc32179251)

[1.5 程序编写&编译 1073766401](#_Toc32179252)

[1.5.1 程序编写 1073766401](#_Toc32179253)

[1.5.2 程序的编译 1073766401](#_Toc32179254)

[1.6 基本数据类型 1073766401](#_Toc32179255)

[1.6.1 整型 1073766401](#_Toc32179256)

[1.6.2 字符型 1073766401](#_Toc32179257)

[1.6.3 浮点型 1073766401](#_Toc32179258)

[1.7 常量与变量 1073766401](#_Toc32179259)

[1.7.1 常量 1073766401](#_Toc32179260)

[1.7.2 变量 1073766401](#_Toc32179261)

[1.8 输入与输出函数 1073766401](#_Toc32179262)

[1.8.1 输出函数 1073766401](#_Toc32179263)

[1.8.2 输入函数 1073766401](#_Toc32179264)

# C语言基础

## C语言概述

### 何谓C语言

语言：人与人沟通的方式

普通话：中国人与中国人进行交流的一种语言

C语言：人与计算机进行交流的一种语言

计算机：只能识别机器语言(二进制文件0 1 )

C语言(人) 🡪 编译(翻译) 🡪 机器语言(计算机)

面向对象 🡪 C++（直接调用别人函数）

面向过程 🡪 C

低级语言：汇编(转化效率高 计算机)

高级语言：C语言(转化效率低 人)

### C语言的发展史

B语言 🡪 C语言

刚出现C语言，无规则 🡪 无规矩不成方圆

在1983年成立C语言委员会

在1989年才出现了C语言的第一条标准 🡪 C89标准

在1999年才出现了C语言的第二条标准 🡪 C99标准

在2011年才出现了C语言的第三条标准 🡪 C11标准

### C语言特点

1. **运算符丰富**

运算：按照某种规则操作数据

算术运算符：+ - \* / %(取余) 10/3=3(取整) 10%3=1(取余)

关系运算符：> < >= <= ==(等于) !=(不等于) 10 <= X <= 20

逻辑运算符：&&(逻辑与) ||(逻辑或) !(逻辑非) 10 <= X && X <= 20

单目运算符：++ --

….

1. **数据类型丰富**

在C语言中分为4大类

基本数据类型：整型int、字符型char、浮点型(实型)(单精度flaot和双精度double)

构造数据类型：数组类型

存50个学生年龄(整数) 🡪 需要50个整型变量的空间来存

int age1, age2….age50; 繁琐、代码量大

int age[50];//相当于定义50个整型变量空间，这些空间地址连续

只能存储数据类型相同数据 使用之前空间大小已经确定了(空间不足和空间浪费)

结构体类型：可以存储不同数据类型的数据和相同数据类型的数据

共用体、枚举

链表：合理的利用空间资源

指针类型：数据类型 \*

int \*p;//定义指针变量 🡪 存地址 指针变量只存地址

int age;//定义整型变量 🡪 存数据

char age;//定义字符型变量 🡪 存数据

float age;//定义了浮点型变量 🡪 存数据

空类型：void

1. **能对位、字节、地址进行操作，能直接访问硬件**

位(bit)：计算机最小的存储单位

字节(byte)：计算机最常用的单位

1byte = 8bit

1 kb = 1024byte

1 mb = 1024kb

1 gb = 1024mb

1 tb = 1024 gb

地址：在计算机内存中，系统会为每一个字节分配一个编号，这个以16进制表示，这个就是地址

1. **结构化编程语言**

三种结构+4条跳转语言

1. **转换效率高**

只比汇编低10-20%

1. **可编程的自由度大，每个人都有自己的风格**
2. **可移植性高**

## 搭建开发环境

### GCC编译器

GNU公司旗下的一个产品

当前的这个编译器有不同的版本：windows、Linux、Unix(mac)

### GCC编译器安装

1. 找到自己用来安装软件的磁盘

例如：D盘

1. 在使用的磁盘下面新建一个英文命名的文件夹

在D盘下面建立：gcc

1. 将安装包拷贝到gcc文件夹下，解压到当前文件夹，得到mingw这个文件夹
2. 进入mingw文件夹，进入bin文件夹
3. 拷贝路径



拷贝

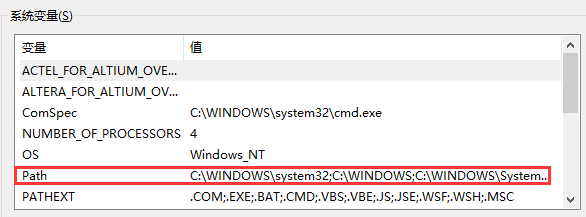
1. 点击此电脑右键，选择“属性”；



1. 选择“选择高级系统设置”



1. 选择“环境变量”，找到系统环境变量



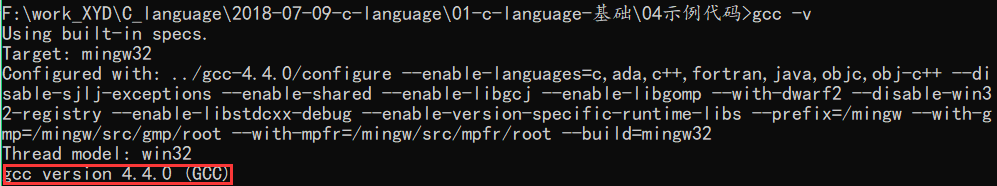
找到最后这个坐标位置。如果后面没有分号加上分号，否则不需要加分号。将刚刚拷贝的路径粘贴到此处。注意：分号是英文的

所有点击“确定”。

软件的安装到此结束。

### GCC编译器验证是否安装

1. Windows+r键输入cmd或者直接找到cmd
2. 在命令框中输入gcc -v
3. 如果出现如下信息表示安装成功

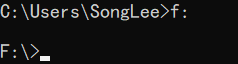


## CMD使用方法

### 切换盘符

盘符名加冒号；例如：f:

写好以后敲回车键即可。



### 进入文件夹命令

cd [可以是一个文件夹，连续的多个文件夹]；写完以后敲回车键。

cd 与文件夹之间有空格键。

cd 后面接的文件夹名字可以通过Tab进行切换

cd ..表示返回上级文件夹

### 查看文件夹名或文件名

dir命令

### 清除命令行显示的信息

cls命令

### 键盘上下键功能

上下键可以选择已经使用过的命令。

### gcc编译命令

编译程序代码方法：

gcc xxxx.c -o xxxx

查看编译结果：

xxxx.exe

## C语言的程序结构

|  |
| --- |
| #include <stdio.h>  int main(void)  {  printf("hello world\r\n");  return 0;  } |

### 头文件包含

#：预处理命令 在编译之前完成的工作

include：英文包含的意思，文件包含命令 🡪 找到<stdio.h>，并把这个文件复制到本文件中来

<>/””：都是寻找文件方式

<>：只会在系统默认路径下寻找文件 VC6.0：安装路径

“”：先在自定义路径(.c文件所在路径)下寻找文件，找不到才会去系统默认路径下寻找文件

stdio：表示文件名 代表文件的内容

std：standard 标准

i:input

o:output 🡪 标准输入输出

.h：文件扩展名 🡪 决定文件的属性 头文件：函数声明、宏定义

.txt记事本文件 .jpg图片文件 .mp3 音乐文件

总结：在编译之前，在系统默认路径下找到标准输入输出头文件，并把这个文件的内容复制过来

### 注释

注释：给程序员看方便阅读代码，不需要遵循语法规则

单行注释

// 只有一行 不需要遵循语法规则

多行注释

/\*

…

…

\*/

### 函数注释头

函数注释头：方便程序员查看函数功能

/\*

函数功能：实现函数功能

函数形参：形参列表

返回值：0

备注：算法、难以理解的代码

日期：xxxx年xx月xx日

作者：xxx

版本：V0.0

\*/

### 主函数

在C语言中，主函数都是名是main (错误mian)

格式：返回值类型 main(void)

常见格式

void main(void) 🡪 OK 不是标准写法，但是编译器允许一种写法

int main(void) 🡪 标准的

main(void) 🡪 OK 默认是int

注意：

主函数作为程序的入口，主函数结束，程序也结束

主函数有且只有一个

## 程序编写&编译

### 程序编写

编写过程

1. 明确程序的目标 🡪 实现的功能
2. 程序的框架搭建 🡪 程序思路
3. 编写代码
4. 编译
5. 检查语法错误 🡪 先解决第一个，解决一个再编译一次
6. 对比写出来和实现需要

### 程序的编译

C语言 🡪 编译 🡪 机器语言 编译不是一步：分为4个步骤 脚本：批处理

看编译的过程：需要GCC编译

1. 预处理：在编译之前完成工作，找文件并把文件的内容复制过来

生成的文件01.i 命令：gcc -E 01.c -o 01.i

1. 编译：检查语法错误，生成汇编文件

生成的文件01.s 命令：gcc -S 01.i -o 01.s

1. 汇编：将汇编转化为机器语言(二进制文件)

生成的文件01.o 命令：gcc -c 01.s -o 01.o

1. 链接：将机器语言转化为可执行文件

生成的文件01.exe 命令：gcc 01.o -o 01.exe



## 基本数据类型

### 整型

1. 关键字：int 整数 10 12 100 int a;
2. 分类
3. 按大小之分：short(短整型) int long(长整型)
4. 按修饰符之分：signed(有符号，有正负, 默认) unsigned(无符号，无正负)
5. 存储空间的大小

short：2字节空间 short a;

int：32位及以上占4字节空间 32位以下2字节

long：4字节

1. 存储空间的范围

数据都是以补码的形式进行存储的 15

对于正数：原码 = 反码 = 补码

对于负数：-11(8bit)

最高位符号位：0正数 1表示负数

原码：二进制编码 1000 1011

反码：数据位取反 1111 0100

补码：反码+1 1111 0101

unsigned short a; 2字节 16bit 1000000

存储格式：所有位都是数据位

取值范围 0 - 65535

最大值：所有位都为1 1111 1111 1111 1111 65535

最小值：所有位都为0 0000 0000 0000 0000 0

signed short a; 2字节 16bit

存储格式：最高位符号位+15位数据位

取值范围：-32768 - 32767

最大值：符号位为0，数据位都为1；0111 1111 1111 1111 32,767‬

最小值：符号位为1，数据位都为0；1000 0000 0000 0000 -32768

unsigned int a; 0 – 40几亿

signed int a; -20几亿- +20几亿

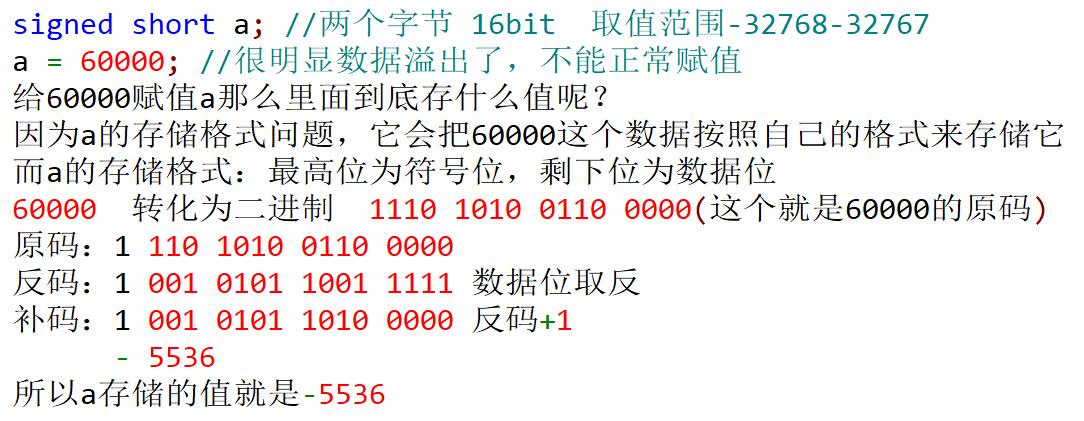
1. 数据的溢出

不是给什么数据就能存什么数据

存60000值是什么？

作业

signed short a; -32768 – 32767 给60000发生溢出，到底存什么值？



### 字符型

1. 用单引号括起来的数据： ‘1’ ‘a’ ‘\n’ ‘\012’ 八进制：0开头数据都是8进制
2. 关键字：char
3. 分类

修饰符之分：signed(有符号，正负) unsigned(无符号，无正负)

1. 存储空间的大小

char 1字节

1. 存储空间的范围

字符型数据的存储格式和整型数据的存储格式一样。

signed char a; 1字节 8bit

存储格式：最高位符号位+7位数据位

取值范围：-128 – 127

unsigned char a;1字节 8bit

存储格式：所有位都为数据位

取值范围：0 - 255

1. 字符型数据的存储

用特定的整型数据对应特定的字符型数据，整型数据和字符型数据都是一一对应的关系。

美国标准信息交换表(ASCII码表)

1. 字符数据的分类

功能字符：只体现功能，不能显示

显示字符：能在标准输出(屏幕)能显示出来

功能字符:

‘\n’ 🡪 换行

‘\r’ 🡪 回车

‘\r’’\n’ 🡪 标准换行(单片机里面常用)

显示字符

‘ ’ 🡪 空格 32

‘0’ – ‘9’ 🡪 48 - 57

‘A’ – ‘Z’ 🡪 65 - 90

‘a’ – ‘z’ 🡪 97 – 122

1. 字符的应用
2. 如何将字符’0’转化为十进制0

‘0’ – ‘0’ = 0

‘9’ – ‘0’ = 9

1. 如何将小写’a’转换为大写字母’A’

‘a’ – 32 = ‘A’

1. 字符可以进行运算？

可以

### 浮点型

也称为实型 float a; 小数

1. 关键字

单精度：flaot

双精度：double

可不可以用修饰符修饰？不可以

浮点型数据可以存负数吗？可以存正负

1. 存储空间的大小

flaot：4字节

double：8字节

1. 存储空间的范围

不用去记，范围很大

浮点数据的存储格式与整型数据的存储格式不一样

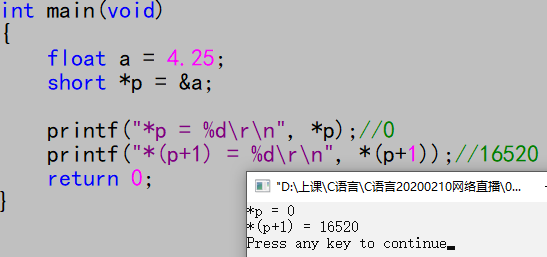
float：4字节 32bit

存储格式：最高位符号位+(127+指数值)指数位8bit+23位小数位

4.25 十进制小数转化二进制

4.25 🡪 二进制格式 100.01 🡪 指数格式1.0001 \* 2^2

0100 0000 1000 1000 0000 0000 0000 0000



double：8字节 64bit

存储格式：最高位符号位+(127+指数值)11bit+52bit小数位

## 常量与变量

### 常量

1. 何谓常量？

在程序运行过程中，不会改变的量，就称为常量 只能进行读操作不能进行写操作

1. 常量的数据类型固定的，不需要指定 10 ‘a’ 12.34(小数)

整型常量数据默认数据类型是： int类型

字符型常量数据默认数据类型是：char类型

浮点型常量数据默认数据类型是：double类型

1. 字符串(不属于数据类型)
2. 用双引号引起来 “1”= ‘1’’\0’
3. 多个字符和’\0’组成
4. 字符串都是以’\0’结尾，’\0’不显示出来
5. 字符串的空间大小会比多个字符多一个 “hello”一共占多大空间？6个字节 sizeof

### 变量

1. 何谓变量（空间）

在程序运行过程中，可能发生改变的量，就为变量 可读可写

1. 变量的定义和声明(对于变量定义也叫声明)

格式：数据类型 变量名 char a; int b; float c;

数据类型：决定空间大小和存储格式

变量名：满足标识符的命名规则，具有含义

int a; int age; int nianling;

1. 标识符命名规则

组成：由数字、字母、下划线和$组成

不能以数字开头

有大小写之分

不能为关键字

1. 变量初始化和赋值

初始化：定义变量的同时进行赋值 int age = 18;//初始化

赋值：先定义变量，再进行赋值 int age; age = 18;

1. 全局变量和局部变量

全局变量？ 在函数外定义的变量

局部变量？ 在函数内定义的变量

局部变量 🡪 如果不进行初始化，默认是什么值？随机值(垃圾值)

生命周期：从定义这个变量开始到函数结束

作用域(作用的区域)：从定义变量到函数为止

全局变量 🡪 如果不进行初始化，默认是什么值？0

声明周期：从定义这个变量开始到程序结束

作用域：从定义这个变量开始到以下所有部分

1. 变量用法

读写

int age; //4字节 定义一个整型变量age

在计算机内存中为你分配了4字节连续空间，以4字节中低地址作为age空间的首地址

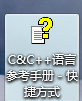
写：把变量”=”左边 a = 10；

读：把变量”=”右边 b = a;

## 输入与输出函数

在C语言中函数可以分为三类

1. 主函数
2. 自定义函数(自己写的函数，也叫子函数)
3. 库函数(别人写好的，自己使用即可) scanf printf



### 输出函数

函数原型：int printf( const char \*format, ... );

函数功能：将数据按照格式输出到屏幕(标准输出)上

函数形参：

const char \*format：格式控制字符串（格式控制字符和非格式控制字符(给什么就显示什么)）

printf(“a = %d\r\n”, a);

printf(“hello world”);

在格式控制字符串中，有多少格式控制字符，就需要多少个参数

格式控制字符

%d：以十进制的格式输出

%c：以字符的格式输出

%s：以字符串的格式输出

%f：以单精度的浮点型格式输出 默认保留小数点后6位

%lf：以双精度的浮点型格式输出 默认保留小数点后6位

%o：以八进制的格式输出

%x：以十六进制的格式输出 0x开头

在%和x可以插入# 如果x是小写出来的就是小写的16进制

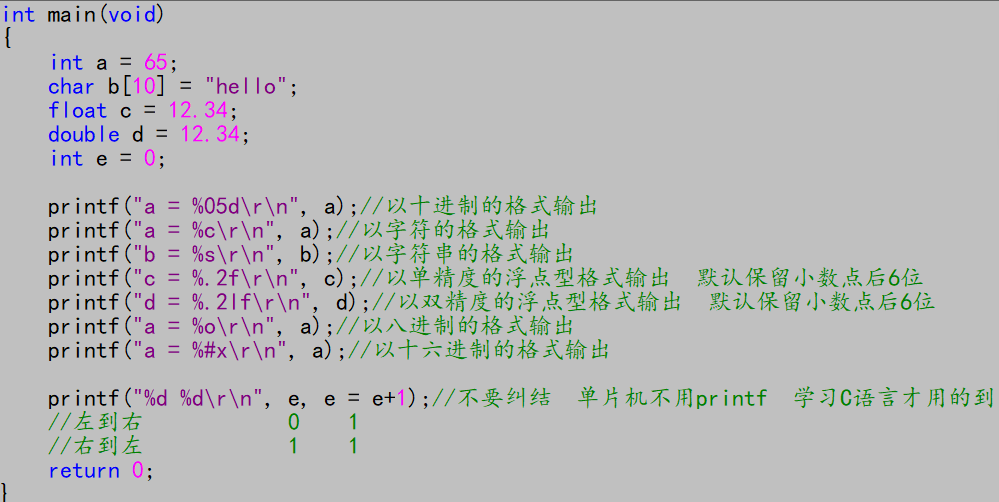
在%和字符之间可以插入数据

插入的正数：右对齐

插入的小数：左对齐(默认)

插入0：补全位数

插入小数：决定浮点型数据小数点位数



### 输入函数

1. 函数原型：int scanf( const char \*format, ... );
2. 函数功能：将从键盘中得到的字符按照格式输入对应的变量空间中
3. 函数形参：

const char \*format：格式控制字符串(格式控制字符和非格式控制字符)

scanf(“a = %d”, &a); &：取地址符

怎么输入？a = 10 格式控制字符串里面有什么非格式控制字符就需要输入对应非格式控制字符

建议：在格式控制字符串里面不要有非格控制字符，格式控制字符之间用空格隔开(有多个格式控制字符)

scanf(“%d”, &a);

格式控制字符

%d：以十进制的格式获取

%c：以字符的格式获取

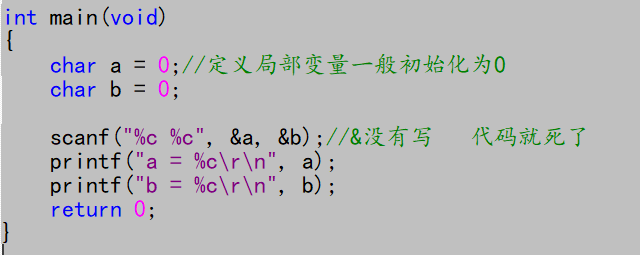
%s：以字符串的格式获取

%f：以单精度的浮点型格式获取

%lf：以双精度的浮点型格式获取

%o：以八进制的形式获取

%x：以十六进制的格式获取



1. 补充sizeof 统计空间大小

是不是函数：不是函数是关键字

