# 简单C程序

Dos窗口打印输出字符串

第一步 编写代码 保存为hello.c文件

#include <stdio.h> //使用printf库函数之前，必须先申明此头：和java导包一样

int main(){ //必须要有主函数

printf("sdfsd\n"); //printf是c语言的库函数：用来打印字符串

printf(”%d”,3); //整数的输出：java直接+

return 0;

}

第二步 编译a.c 前提有安装编译器：mingw64\_8.1配置环境path：D:\ mingw64\bin;

在dos命令窗口：gcc hello.c 此时会有一个：a.exe linux:是a ；没有扩展名

指定编译后的名字：gcc -o b hello.c 就会生成 b.exe

第三步 运行

dos命令窗口：直接运行编译好的文件名称：a.exe

# 常量

宏常量：关键字#define

#define MAX 10 //定义一个宏常量，值为10，常量不能改变

#include <stdio.h>

//1、#开头的语句是预处理语句，无需分号结束

//2、定义一个宏定义的标示符MAX， 它代表100， MAX它是常量

//3、宏定义以后的代码都可以使用MAX

//4、在后面出现的MAX，预处理都会替换为100

#define MAX 100

#define STRING ”hello world”//定义一个字符串常量

int main(int argc, char \*argv[])

{

//10 = 1;

//MAX = 1; //MAX它是常量， 不允许修改

int b = MAX;

int d;

d = 100; //ok, 变量可以修改

int e = c; //把c的值给e， c本身没有变量

return 0;

}

const 常量

//const是一个关键字，作用是修饰一个变量为只读

const int c = 10;

//c = 100; //err, c只读，不允许修改

# C语言库

## stdlib.h

1、运行dos命令：如命令’notepad’是打开记事本

#include <stdlib.h>

int main(){

system("notepad"); //库函数：进行打开doc命令：记事本 但是不会关闭dos窗口

system("start notepad"); //打开记事本，关闭dos窗口

return 0;

}

不弹doc窗口：需导包：<Windows.h> 库函数：WinExec(” notepad”,SW\_NORMAL)

system("notepad"):进行编译后就可以双击打开a.exe,不过会同时打开都是命令窗口和记事本

同理可以运行自己写的C语言编译后exe文件：接收a.exe主函数返回i的值

注意：linux处理会有不同的结果，window下是正常的结果

第一步 a.c文件内容主函数有int返回值：

#include <stdio.h>

int main(){

int i=666;

printf("这个i的值是：%d",i); //打印i的值

return i; //主函数 返回i的值

}

第二步 b.c的内容：使用函数执行a.exe内容并接收返回值

#include <stdio.h>

#include <stdlib.h>

int main(){

int e;

e=system("a"); //执行a.c编译后exe文件并接收a的主函数返回的内容

printf("a.exe文件中主函数返回的值是：%d",e); //window正常接收

printf("a.exe文件中主函数返回的值是：%d",e/0x100); //linux系统下需要除于16进制的100

return 0;

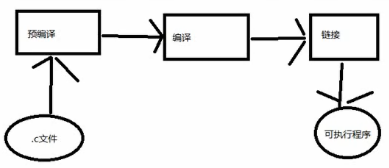
}

第三步 执行编译b.c文件：gcc -o b b.c，在dos窗口运行b.exe就会打印：a.exe 和b.exe中的内容输出

老师提到了：POSIX标准的C/C++库，尽量不要用windows移植麻烦

# GCC

gcc 编译过程



预编译参数：gcc -E :

gcc –E –o aa ee:这里意思是把aa.c文件预编译成ee.c文件

打开文件ee.c 文件发现：1、内容很长，自己写的代码在最后面#include <stdio.h>没有了：这里因为把stdio.h全部内容替换到ee.c中，2、没用的注释都删除掉了

汇编参数：-S 还可以打开文件

就是讲c语言转化为汇编语言(老师演示了vs中调试：打断点----运行后调试----三个正方形的图标)

int c; a=1;b=2;c=a+b; 这几句代码的汇编语言是：

-asm{

mov a,1 //把1移到a中

mov b,2

mov eax, a //把a放到寄存器中

add eax, b //添加b到寄存器中

mov c, eax //从eax中获取c的值

}

编译参数：-c编译 将代码编译为二进制的机器指令

链接 通过链接器将一个个目标文件（或许还会有库文件）链接在一起生成一个完整的可执行程序

gcc 没有任何参数代表链接

参数 o 代表指定名称：

gcc -o cc a.c :把a.c编译成名称为cc的文件，linux：就是这个名字没有扩展名，window：cc.exe