

## 1.#include<> 和 #include" 的区别

- 1 #include<>只会查找系统的头文件
- 2 #include"" 不仅查找系统的头文件也会查找自己编写的头文件

## 2.int main()主函数带参数

- 1 `int main(int argc, char* argv[])`
- 2 `argc`: 是参数的个数, 默认是一个
- 3 `argv`: 是指针数组, 指向各个字符串的首地址, 其中`argv[0]`存的是可执行文件的文件名的首地址

## 3.return 0

- 1 `return 0;`
- 2 `return 0` 告诉系统该程序执行到此处正常结束, 是程序正常结束的标志。

## 4.基本数据类型:

整型				浮点型	
char	short	int	long	float	double
1个字节	2个字节	4个字节	4/8个字节	8个字节	8个字节
-128-127		-2147483648 -2147483648			

## 5.内存

内存的基本操作单位是字节, 内存访问分为直接访问和间接访问, 内存的基本操作方式有读, 写, 取地址。

## 6.内存的划分

- (1).栈区: 编译器自动分配, 存放参数的参数值, 局部变量等, 程序结束的时候编译器自动释放。
- (2).堆区: 程序员分配释放, 程序员不释放, 又OS回收。分配方式类似于链表。
- (3).全局区(静态区): 存放全局变量, 静态变量, 常量等。
- (4).字符常量区: 存放字符串。
- (5).代码区: 存放函数体的二进制代码。

## 全局变量和静态变量的区别:

- (a) 若程序由一个源文件构成时, 全局变量与全局静态变量没有区别。
- (b)若程序由多个源文件构成时, 全局变量与全局静态变量不同: 全局静态变量使得该变量成为定义该变量的源文件所独享, 即: 全局静态变量对组成该程序的其它源文件是无效的。

(c)具有外部链接的静态；可以在所有源文件里调用；除了本文件，其他文件可以通过extern的方式引用；

### 堆区和栈区的区别：

(a).管理方式不同：栈区是编译器自动管理的，不需要程序员手动控制，堆区分配和释放大多是程序员控制，容易产生内存碎片。

(b).生长方式不同：栈区是从高地址向低地址扩展的数据结构，是一个连续的内存区域，比如申请a,b两个变量，很可能a的地址大于b的地址。而堆区是从低地址向高地址扩展的数据结构，这是由于系统是用链表来管理内存地址的，自然是不连续的。

(c).分配效率不同：栈区的分配效率高，堆区的分配效率低。