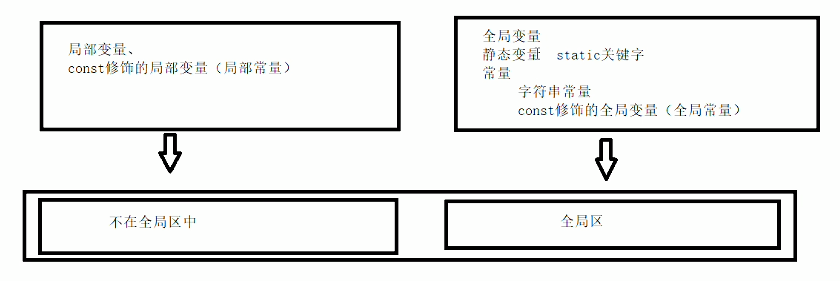
1. 内存分区：

**程序运行前**，即编译生成了exe文件，但还没有运行时

**代码区：**存放CPU执行的机器指令、共享、只读

**全局区：**存放全局变量、静态变量（static）、字符串常量、const修饰的全局变量（全局常量），程序结束时内存自动释放。



栈区：存放局部变量、函数的形参，栈区的数据由编译器管理开辟和释放

堆区（程序运行时）: 利用new在堆区，由程序员动态开辟和释放

1. 引用的本质在c++内部实现是一个常量指针

T& ref = a; 相当于=> T \* const ref = &a;（只能指向变量a的常量指针，可以改变指针指向的内容，但不能改变指针所指的变量）

引用必须初始化；

引用类型的初始值必须是一个对象；eg: int& a = 10; (错误，引用需要一个合法的内存空间)

1. 函数

函数声明和定义只能有一个有默认参数；

占位参数：

函数的返回值不可以作为函数重载的条件；（即，函数仅仅是返回值类型不同，不能重载）

1. 重载

引用作为重载的条件；

void func(int& a);与void func(const int& a);可以重载

函数重载碰到默认参数；

拷贝构造函数

浅拷贝：简单的赋值拷贝操作，编译器默认执行的拷贝构造函数

深拷贝：类中有指针时，在拷贝对象的时候，需在堆区重新开辟一块堆区内存，进行拷贝操作；