

变量作用域

局部变量

- 在一个函数内定义，只在函数范围内有效
- 在复合语句中定义，只在复合语句中有效
- 随着函数调用的结束或复合语句的结束局部变量的声明声明周期也结束
- 如果没有赋初值，内容为随机

静态(static)局部变量

- static局部变量的作用域也是在定义的函数内有效
- static局部变量的生命周期和程序运行周期一样，同时static局部变量的值只初始化一次，但可以赋值多次
- static局部变量若未赋以初值，则由系统自动赋值：数值型变量自动赋初值0，字符型变量赋空字符

全局变量

- 在函数外定义，可被本文件及其它文件中的函数所共用，若其它文件中的函数调用此变量,须用extern声明
- 全局变量的生命周期和程序运行周期一样
- 不同文件的全局变量不可重名

静态(static)全局变量

- 在函数外定义,作用范围被限制在所定义的文件中
- 不同文件静态全局变量可以重名,但作用域不冲突
- static全局变量的生命周期和程序运行周期一样，同时static全局变量的值只初始化一次

内存分布

- 可执行文件 demo (Windows下是 demo.exe)在未运行时，只是提前规划空间，并没有实际占用内存。
- text代码区：存放 CPU 执行的机器指令。通常代码区是可共享的（即另外的执行程序可以调用它），使其可共享的目的是对于频繁被执行的程序，只需要在内存中有一份代码即可。代码区通常是只读的，使其只读的原因是防止程序意外地修改了它的指令。另外，代码区还规划了局部变量的相关信息。
- data全局初始化数据区/静态数据区：该区包含了在程序中明确被初始化的全局变量、已经初始化的静态变量（包括全局静态变量和局部静态变量）和常量数据（如字符串常量）。
- bss未初始化数据区：存入的是全局未初始化变量和未初始化静态变量。未初始化数据区的数据在程序开始执行之前被内核初始化为 0 或者空（NULL）。

- stack栈区：栈是一种先进后出的内存结构，由编译器自动分配释放，存放函数的参数值、返回值、局部变量等。在程序运行过程中实时加载和释放，因此，局部变量的生存周期为申请到释放该段栈空间。
- heap堆区：堆是一个大容器，它的容量要远远大于栈，但没有栈那样先进后出的顺序。用于动态内存分配。堆在内存中位于BSS区和栈区之间。一般由程序员分配和释放，若程序员不释放，程序结束时由操作系统回收。

```
cat@blackcat:~/NOIP$ size demo
   text    data     bss      dec         hex filename
   2345     656     280     3281         cd1 demo
```

快乐刷题

- [P273 组合数](#)
- [P272 回文数个数](#)
- [P262 字符串判等](#)