Linux命令：

|  |  |
| --- | --- |
| cd | Change directory |
| pwd | Print work directory |
| ls | List files |
| cp | Copy files |
| rm | Remove files |
| rmdir | Remove directory |
| ps | Process status |
| man | Manual |
| Sudo | Switch user do |

Linux命令格式：comman [-options] [argument]

* command：命令名，如前面的cd命令，pwd命令，ls命令。
* -options：命令的选项，使用“-”开头的或“–”开头，如前面示例“ls -l”中的-l选项， 命令会根据具体的选项执行不同的操作。使用“-”时一般是选项的简写， 一些选项可以同时使用，“ls -la”等价于“ls -l -a”，即同时使用“l”和“a”选项。 而“–”一般后面带选项的全名，如“ls -a”等价于“ls –all”。
* argument：命令参数，如前面cd命令中以“/home”可“~”参数作为路径名。

环境变量：是在操作系统中一个具有特定名字的对象，当要求系统运行一个程序而没有告诉它程序所在的完整路径时，系统除了在当前目录下寻找此程序外，还会在环境变量中指定的路径去寻找。

编译过程

1.词法分析，2.语法分析，3.中间代码生成，4.优化代码、5.目标代码

代码管理：

1. 把不同功能的代码放在不同的文件中，并且把必要的函数放在对应的头文件中
2. 对软件模块划分层次

**Chap7 Linux应用程序编程基础**

应用程序在内存中分为两个部分：存放代码的代码段和存放数据的数据段

数据段分为栈（stack）和堆（heap）

BSS（uninitialized data segmant）用于存放全局未初始化变量，属于数据段。

栈（stack）：用于存放函数的参数、局部变量等。被调用的函数的参数和返回值被存储在当前程序的栈区，之后再为自身的自动变量和临时变量在栈区分配空间。

栈的管理方式是FILO（first in last out），栈内的数据是同一个方向管理的。

堆（heap）：由程序员管理，利用操作系统提供的分配和释放函数使用堆区的内存。

链表：物理存储单元上非连续、非顺序的存储结构，数据元素的逻辑顺序是通过链表中的**指针链接次序**实现的。

void \*malloc(size\_t size):返回值为分配内存的首地址，分配失败返回NULL，

void free(void \*ptr)释放内存

void \*calloc(size\_t nmemb,size\_t size)分配nmemb个元素，每个元素大小size，总内存空间nmemb\*size，并将内存空间初始化清0（与malloc的区别）

void \*realloc(void \*ptr,size\_t size),调制已经分配的内存空间

Linux中共享文件夹的默认路径为：/mnt/hgfs

共享文件夹它是在Windows也是在Linux下的文件夹，因此我们不要在Linux直接操作共享文件夹的 文件，而是应该将它拷贝到完全属于Linux的地方，比如可以在Linux的桌面上创建一个文件夹， 命名为“work\_dir”,然后通过cp命令或者直接复制粘贴的方式将共享文件夹的文件拷贝到这个 “work\_dir”工作区域文件夹中。