##### Vim回顾

1. 模式：输入、编辑、末行、可视化（v/V）
2. 编辑🡪输入：i/I，a/A，o/O，c/C
3. 编辑下的范围跳转：h/j/k/l，w/b/e，^/$/0，G/#G，(/)，{/},
4. d,y,c,u,ctrl+r
5. 多个编辑器同时编辑同一个文件
6. –o,-O;
7. Vim +# FILE:打开文件，光标直接出现在文件第#行

##### Bash编程

1. Bash进程：命令解释器
2. Shell脚本（.sh结尾）：
   1. 第一行顶格：shebang

#!/bin/bash

#注释信息，描述脚本相关信息

#!/usr/bin/python

其他以#开头的行是注释信息，会被解释器忽略；

* 1. 练习：创建一个组newgroup，iD为8008；创建一个用户为mageedu，id为3306，附加组为newgroup；创建目录/tmp/hellobash；复制/etc/rc.d/init.d/functions至/tmp/hellobash目录中
  2. 过程式编程语言：

1. 顺序执行
2. 选择执行(测试条件，可能会有多个，当测试条件满足时执行对应的分支)
3. 循环执行（将同一代码反复执行多次，循环必须有退出条件，否则陷入死循环）
4. bash –n FILE——作语法测试
5. bash –x FILE——单步执行
6. ./FILE.bash 需要添加权限chmod +x FILE，也可直接bash FILE，不用添加权限
7. 变量类型：
   1. 本地、环境（export——导出）、局部、位置（$1,$2，…,${10}，$0是第一个参数）、特殊
   2. 练习：写一个脚本，能接受一个参数作为用户名，此脚本可创建此用户，并且其密码同用户名
   3. 特殊变量：
8. $?——上一个命令的状态返回值
9. $#——传递给脚本参数的个数
10. $\*或$@——引用传递给脚本的所有参数
11. $1+shift（轮替）：可引用所有参数，每次shift可以轮替下一个参数（shift [n]）
12. Bash循环语句：
    1. While：
    2. Until：
    3. For：遍历有限的元素列表
13. 语句格式：

for VAR\_NAME in LIST

do

循环体

done

1. LIST：列表，中间包括一个或多个元素
2. 练习：创建10个用户，分别为user301-user310
3. 生成数值列表：
   1. {startnum..endnum}——例如{1..100}
   2. Seq命令引用：seq三种命令参数，man seq
4. 练习：创建目录/tmp/dir-当前日期时间：例如/tmp/dir-20140707-155503,在此目录中创建10个空文件，分别为file1到file10
5. 练习
   1. 写一个脚本

创建用户tuser1-tuser9；

创建目录/tmp/dir-当前日期时间；

在/tmp/dir-当前日期时间目录中创建9个空文件file101-file109；

将file101的属主改为tuser1，以此类推，一直将file109的属主改为tuser9

* 1. 写一个脚本

脚本可以接受一个以上的文件路径作为参数；

显示每个文件所拥有的行数；

显示本地共对多少个文件执行了行数统计

* 1. 写一个脚本

1. 显示/etc/passwd文件中位于文件的第偶数行的用户名；并显示共有多少个这样的用户