



北京邮电大学

Beijing University of Posts and Telecommunications



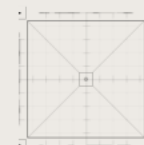
表达式运算

蒋砚军 北京邮电大学计算机学院

► 表达式计算



- shell不支持除字符串以外的数据类型，不支持加减乘除等算数运算和关于字符串的正则表达式运算
- 需要这些功能，借助于shell之外的可执行程序/usr/bin/expr实现
- 有的shell（包括bash）为了提高执行效率，提供内部命令版本的echo,printf,expr,test,[等命令，但这仅仅是一种性能优化措施。只依赖外部命令完全可以实现
- bash提供的这种机制使得程序员可以根据需要自己设计更多更方便的运算



► expr命令：算术运算、关系运算、逻辑运算、正则表达式运算



■ 括号

()

■ 算数运算(5种)

+ - * / %

■ 关系运算(6种)

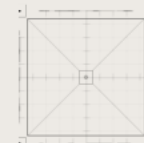
> >= < <= = !=

■ 逻辑运算(2种)

| &

■ 正则表达式运算

:



► 使用expr命令：空格与转义



■ 注意

- ◆ 算术运算、关系运算和逻辑运算，shell脚本用到的时候不是很多
- ◆ 应该转义的地方必须加反斜线转义
- ◆ 应该有空格的地方不允许漏掉

■ 例 $a=2;b=8;c=10$;求 $x=a*(b+c)$ ， $y=a+4<b$ 并且 c 不等于8

```
x=`expr $a \* \( $b + $c \)`
```

```
y=`expr \( $a + 4 \< $b \) \& \( $c != 8 \)`
```

```
x=`expr $a '*' '(' $b + $c ')`
```

```
y=`expr '(' $a + 4 '<' $b ')' '&' '(' $c != 8 ')`
```

► 正则表达式运算



■ 用法: `expr string : pattern`

- ◆ 正则表达式`pattern`匹配字符串`string`，打印匹配长度
- ◆ `pattern`中用`\`(和`\`)括起一部分，能匹配时打印括号内能匹配的部分，否则为空字符串

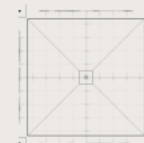
■ 举例

`expr 123 : "[0-9]*"` 结果为3

`expr A123 : "[0-9]*"` 结果为0

`expr "$unit" : ".*"` 返回变量`unit`的长度

`expr `pwd` : '.*\/\([^\/]*\)'` 截取路径名的最后一个分量



► 正则表达式运算：从标准输出中抽取数据



```
ping -c 1 -w 1 192.168.0.1命令成功时输出如下（ping一次超时时间1秒）  
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.  
64 bytes from 192.168.0.1: icmp_seq=0 ttl=64 time=0.806 ms
```

```
--- 192.168.0.1 ping statistics ---  
1 packets transmitted, 1 received, 0% packet loss, time 0ms  
rtt min/avg/max/mdev = 0.806/0.806/0.806/0.000 ms, pipe 2
```

从中提取的RTT时间0.806ms（注意\$str两侧的引号是ping成功或失败两种情况下必须的）

```
str=`ping -c 1 192.168.0.1 | grep from`  
expr "$str" : '^.* time=\([0-9.]*\) ms$'
```



北京邮电大学

Beijing University of Posts and Telecommunications



谢谢