

Exit status, command test, flow control.

Jan Trdlička

Department of Computer Systems
Faculty of Information Technology
Czech Technical University in Prague

trdlicka@fit.cvut.cz

October 23, 2018

Contents

- 1 Test 1
- 2 Exit status
- 3 Command test and if
- 4 Loop for
- 5 Homework

Test 1

- Before test
 - Boot the local PC into the Progtest image.
 - Open two terminals.
 - The first terminal for CLI.
 - The second terminal for UNIX manual pages.
- Recommendation
 - Read question carefully.
 - Try to find a correct solution of the question.
 - Verify your solution in CLI.
 - Write down your solution into a paper test.
- Test conditions
 - Work alone!!!
 - Use only UNIX manual pages, no other materials!!!
 - Violations of test conditions means a zero rating.

- What is the exit status?
- What is the meaning of exit status of the following commands?

- ```
PASS=/etc/passwd
grep ^root: "$PASS" ; echo $?
```

- ```
grep ^rooooot: "$PASS" ; echo $?
```

- ```
grep ^root: /etc/foo ; echo $?
```

- ```
~/.bash_history ; echo $?
```

- What is the meaning of exit status of the following commands?

- `winzip ; echo $?`

- `ls -lR / ^C ; echo $?`

Command test and if

- Create a shell script `list1.bash` that requires one argument.
 - If the argument is missing or if there are more arguments, then the script prints the following error and exits with exit status 1.

```
Usage: ./list1.bash directory_name
```

- If the argument is a name of a directory and the directory is readable, then the scripts lists its contents by command `ls -la`. Otherwise the script prints the following error and exits with exit status 2.

```
./list1.bash: "foo" is not readable directory
```

- Change the access permissions of the script by the following command

```
chmod 755 list1.bash
```

- Run the script by the following command

```
./list1.bash
```

Command test and if

Loop for

- Modify previous shell script such that for every regular readable file in the directory it will print file type (output from command file).
- Example of script output

```
$> ../list2.bash /usr/bin/ | head
```

```
/usr/bin/[: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked  
/usr/bin/4i1toppm: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically  
/usr/bin/7z: POSIX shell script, ASCII text executable  
/usr/bin/7za: POSIX shell script, ASCII text executable  
/usr/bin/a2p: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically  
/usr/bin/aa-easyprof: Python script, ASCII text executable  
/usr/bin/abs2rel: Lua script, ASCII text executable  
/usr/bin/aclocal: awk script, ASCII text  
/usr/bin/aclocal-1.13: awk script, ASCII text  
/usr/bin/aconnect: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically
```


Loop for

Homework

- Create a shell script `list3.bash` that requires one argument.
 - If the argument is missing or if there are more arguments, then the script prints the following error and exits with exit status 1.

```
Usage: ./list3.bash directory_name
```

- If the argument is a name of a directory and the directory is readable, then the scripts counts the number of subdirectories of this directory and prints this information in the following format:

```
$> ./list3.bash /etc/init.d/rc0.d/  
/etc/init.d/rc0.d/: no subdirectory
```

```
$> ./list3.bash /usr/bin/  
/usr/bin/: 1 subdirectory
```

```
$> ./list3.bash /etc  
/etc: 134 subdirectories
```