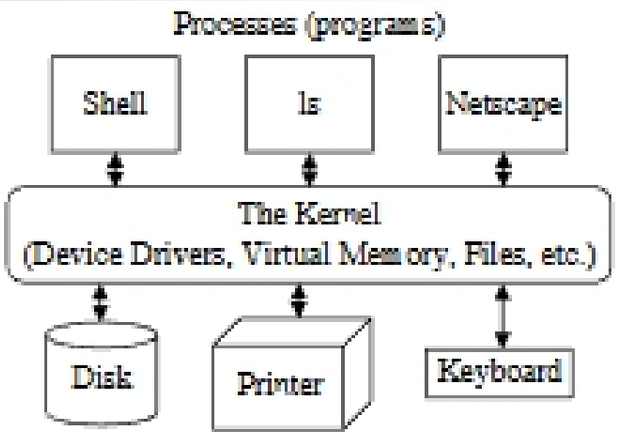
Lecture 3: Unix Shell, Pattern Matching, Regular Expressions

Kenneth M. Anderson
Software Methods and Tools
CSCI 3308 - Fall Semester, 2004

Today's Lecture

- Review Lab 0's info on the shell
- Talk briefly about Lab 1
- Discuss pattern matching
- Discuss regular expressions

Unix Architecture (simplified)



The Unix Shell

- A shell is a program that presents the user with an interpreted programming environment; there are many many shells!
- It provides
  - Variables and built-in commands
  - The ability to execute external commands (e.g. programs)
  - The ability to redirect input and output
  - Shortcuts such as aliases and wildcards
- In this class we are going to be using tcsh
  - Its an extended version of csh (The C Shell); the tcsh author added the "t" after adding features from the TENEX and TOP-10s operating systems to the vanilla C shell
- The purpose behind csh was to emulate the syntax and operators of the C programming language

Variables

% **set x = ken**
% **echo x**
x
% **echo $x**
ken
% **set y = (bananas apples kiwi)**

- This creates a 1-based array (first element indexed with a 1)
- An array is separated by spaces (which can cause problems) and surrounded by parentheses
- What happens if you leave the parentheses out?

- Certain constructs can take advantage of an array

%**echo $y[2]**
apples
%**foreach fruit ($y)**
foreach? **echo $fruit**
foreach? **end**
bananas
apples
kiwi
%**set y[2] = oranges**
%**echo $y**
bananas apples kiwi

More on Spaces

- Arrays are often used to iterate over the contents of directories in the file system
- Since the space character is used as a delimiter for arrays, you need to watch out for spaces that appear in file and directory names
  - See example next slide

Example Directory Structure

- Tenure Review/
  - Tenure Talk
  - Tenure Demo

```
%set z = (`find Tenure\ Review -type d -print`)
%echo $z
Tenure Review Tenure Review/Tenure Talk …
%echo $z[1]
Tenure
%echo $z[2]
Review
```

- This is not what we want!

How to fix?

- Use the shell "quoting" mechanism
- Already saw one example when I used the string "Tenure\ Review" in the find command
  - The backslash "escapes" the space and allows the two words to be treated as a single directory name
- You will learn more about the quoting mechanism in lab; In addition, there is a lot of information about quoting in your reference textbook

Example Revisited

- Tenure Review/
  - Tenure Talk
  - Tenure Demo

```
%set z = (`find tenure\ review -type d -print`)
%echo $z
tenure review tenure review/tenure talk …
%echo $z[1]
tenure review
%echo $z[2]
tenure review/tenure talk
```

- Much better!

Math

- "set" treats the value as a string

```
% set x = (2 + 3)
% echo $x
    2 + 3
% echo $x[2]
    +
```

% Use "@" to do math;

- Note: the space between the "@" and the variable is **REQUIRED**

```
% @ x = (2 + 3)
% echo $x
    5
```

- tcsh supports most of C's expression operators (such as plus, minus, multiply, divide, less than, greater than, equal, etc.)

  + - * / > < == >= <= && || ! ++ -- += -= *= /=

Input/Output Redirection

- tcsh can redirect input and output
  - it can also redirect error output (not shown)

```
% date
    Sun Aug 20 11:11:10 MDT 2000
% date > today
% more today
    Sun Aug 20 11:11:14 MDT 2000
% rev < today
    0002 TDM 41:11:11 02 guA nuS
```


Control Flow Constructs

- Conditional

```
if ($x > 3) then
    echo true
else
    echo false
endif
```

- Iteration

```
while !($done)
    …
end

foreach directory (bin build lib)
    mkdir $directory
end
```