Today

- Regular Expressions
- Grep

1

Review

- What does this command do?   `mv 0{8,7}-for_functions`
- How do you refer to parameters in scripts?
- How can you extract the name of a file from a path?
  - How can you extract the directories (only, not the file name) from a path?
- How can you find out how much disk space is being used by files?
- Often, what do the options –r (sometimes –R) and –h mean in commands?
- What is the syntax of a *for* loop in bash?
- What are the permissions available for Unix? To whom can those permissions be assigned?
  - What is the command to change a file's permissions? Owner? Group?
- What is a regular expression?
  - What are some of things you can specify in regular expressions? How do you represent them?

2

Review: What Is a Regular Expression?

- A ***regular expression*** (***regex***) describes a set of possible input strings

  - ➤ We compare strings to the regular expression to see if they *match*
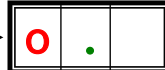
3

Regular Expressions

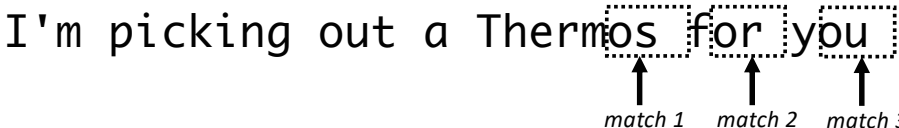- The **.** regular expression can be used to match any character.

regular expression ⟶ | **o** | **.** | |

```
I'm picking out a Thermos for you
```
　　　　　　　　　　match 1　　match 2　　match 3

4

Regular Expressions

- Most of what we went through can be used in commands, like `ls, cp, rm` (be careful!), …
  - ➢ I test the `rm` command with `ls` first
- Practice
  - ➢ List the files that begin with D
  - ➢ List that files that end in .java
  - ➢ List the files that begin with D or d
  - ➢ List the files that begin with a, b, c, or d and end in .py

5

Anchors

- Anchors are used to match at the beginning or end of a line (or both)
- `^` means beginning of the line
- `$` means end of the line

6

regular expression ⟶ | ^ | b | [eor] | a | t |

beat brat boat

↑
match

---

regular expression ⟶ | b | [eor] | a | t | $ |

beat brat boat

↑
match

What would these match to?

^word$

^$

Jan 28, 2022   Sprenkle - CSCI397

7

Special Characters

● ***** : defines **zero or more** occurrences of the *single* regular expression preceding it.

● **+** : defines **one or more** occurrences of the *single* regular expression preceding it.

● **?** : defines **zero or one** occurrences of the *single* regular expression preceding it.

● **|** : defines *or* of the *single* regular expression before and after it

Jan 28, 2022   Sprenkle - CSCI397   8

8

regular expression → | y | a | * | y |

I got mail, yaaaaaaaaaay!

match

regular expression → | z | o | * | z |

This is the best pizza in a cup ever.

match

.* Match 0 or more of *any* character

Match length

- A match will be the longest string that satisfies the regular expression.

regular expression → | a | . | * | e |

scrapple from the apple.

no   *no*   *yes*

Practical Regex Examples

- HTML headers <h1> <H1> <h2> …
  - ➤ `<[hH][1-6]>`
  - ➤ (current standard is lower case h)
- Variable names in C/Python/Java (more constraints but…)
  - ➤ `[a-zA-Z_][a-zA-Z_0-9]*`
- Dollar amount with optional cents (
  - ➤ `\$[0-9]+(\.[0-9][0-9])?`
- Time of day
  - ➤ `(1[012]|[1-9]):[0-5][0-9] (am|pm)`

11

The etymology: https://www.youtube.com/watch?v=NTfOnGZUZDk

GREP

Regular expression tester for grep: https://www.regextester.com/

12

6

grep (global regular expression print)

```
grep [-option] pattern [filename]
```

Option	Meaning
-h	Do not display filenames
-i	Ignore case
-l	List only filenames containing matching lines
-n	Precede matching line with its line number
-v	Select non-matching lines
-x	Match whole line only
-E	Extended regular expression pattern

Or use egrep

Jan 28, 2022   Sprenkle - CSCI397   13

13

grep: Backreferences

- *Backreferences* allow us to refer to a match that was made earlier in a regex
  - ➢ \n : backreference specifier, where n is a number
  - ➢ Looks for nth subexpression
- Example: HTML Tags
  - ➢ `<h[1-6]>.*</h[1-6]>` is not good enough to match html headers, since it matches `<h1>Hello world</h3>`
  - ➢ `<h\([1-6]\)>.*</h\1>` matches what we were trying to match before.

Jan 28, 2022   Sprenkle - CSCI397   14

14

grep: Backreference Examples

- To find if the first word of a line is the same as the last:
  - `^\([[:alpha:]]\{1,\}\) .* \1$`
  - `\([[:alpha:]]\{1,\}\)` matches 1 or more letters
- Another example:
  - `"Mr \(dog\|cat\) came home to Mrs \1 and they went to visit Mr \(dog\|cat\) and Mrs \2 to discuss the meaning of life"`

> What text should this match?

15

grep Examples

- `grep 'men' greptest`
- `grep 'fo*' greptest`
- `grep -E 'fo+' greptest`
  - `egrep 'fo+' greptest`
- `grep -n '[Tt]he' greptest`
- `grep 'The' greptest`

"typically patterns should be quoted in shell command"

16

Looking Ahead

- Assignment 0 – due next Friday
- Guest speaker on Wednesday
  - ➢ Write up of talk will be due on Friday too

Sprenkle - CSCI397   17

17