IT441

Network Services Administration

Handling Text:
Strings

DRAFT

Pipes in Perl

- What is a pipe?
- Why do we use pipes?

- In Perl, we implement pipes using the OPEN statement
  - `open (FH, -|, 'perl sort2.pl gettysburg.txt');`
    - What will this statement do?
    - It will start a Perl program `sort2.pl` using the file `gettysburg.txt` and send the output from `sort2.pl` into our program under the filehandle `<FH>`

<u>Pipes in Perl</u>

- `-|` brings the output of the other program into our program via the filehandle
- `|-` takes the output of our program and sends it to the other program using the file handle and a print statement

File Tests

- Before we do anything we often would like to know the status of the file or directory we are working with.

- We can do this with the following type of test
  - **if (flag "somefile.dat") {action}**

    or

  - **if (flag $somefile) {action}**

File Test Flags

- **`-e`**   true if the file exists
- **`-f`**   true if a plain file – not a directory
- **`-d`**   true if file is a directory
- **`-z`**   true if file has zero size
- **`-s`**   true if file has nonzero size -- returns size
- **`-r`**   true if file is readable by you
- **`-w`**   true if file is writeable by you
- **`-x`**   true if file is executable by you
- **`-o`**   true if file is owned by you

This is *Table 8-1* on *page 201* in the textbook

String Processing

- Remember ***strings*** are the basic data type in Perl
- We have already learned one way to process a string.
  - We can use a ***regex*** (a regular expression)
- Remember how the characters in a string are counted
  - The first (left most) character is `0`
  - The last (right most) character is `-1`

- There are other ways to process strings in Perl
- Perl has many built-in functions to process strings.

String Functions

- Some string functions implemented in Perl
  - `length(string)`
    - Use this function to determine the length of the string.
  - `index(string, substring)`
    - Use this function to determine the `0`-based location of the substring in the string. If substring is not found it returns a -1.
  - `rindex(string, substring)`
    - Similar to `index()` but starts from the right-most end.
  - `substr(string, starting-index, length)`
    - This function returns a substring of `length` number of characters starting from `starting-index`.