

EECS 22: Advanced C Programming Assignment 1

Guantao Liu
guantaol@uci.edu

09/22/2016

Lab Sessions

- Lab Sessions
 - Discuss assignments
 - Please read the assignment handout ahead of time.
 - Address questions regarding lecture, concepts and homework
 - Programming
 - Work on the assignment
 - Individual work but discussion is encouraged.
 - Fix program bugs
- Attendance is not mandatory but highly recommended!

Outlines

- Linux Operating System
- EECS Servers
 - zuma.eecs.uci.edu
 - crystalcove.eecs.uci.edu
- SSH Software
- Basic Linux Commands
- VI Text Editor
- Programming Discussion: Collatz conjecture

Linux Operating System

- OS family: Unix-like
- A prominent example of free and open-source software
 - The source code can be assessed, modified and distributed freely



Linux

Login to EECS Servers

- Login to the machines
 - zuma.eecs.uci.edu
 - crystalcove.eecs.uci.edu
 - Use your EECS account and password to login
 - Use your UCInetID and password to retrieve your EECS account
 - <https://newport.eecs.uci.edu/account.py>
 - After logging into a machine, please change your initial password
 - yppasswd

SSH Software

- Use a terminal with SSH protocols (secure shell)
 - Windows: putty, OpenSSH, cygwin
 - Mac: Terminal (Spotlight->terminal)
 - Linux: Terminal (Applications->System Tools ->Terminal)
 - Authorize yourself with username and password
 - Password will not be shown explicitly on the screen.
 - Enable X windows if you need graphic interfaces

Basic Linux Commands

- Linux shell prints command prompt awaiting input

ls	list files
cd	change working directory
pwd	print working directory
mkdir	make directory
mv	rename/move files
cp	copy files
rm	remove files
rmdir	remove directory
cat	print the content of a file
more	print the content of a file, one screen at a time
echo	print the arguments on the rest of the command line

Basic Linux Commands

- Linux working environment: texture-based
- One way to find the usage and options of a Linux command
 - Linux command manual page: `man rm`
- Directory paths

.	current directory
..	one level higher
~	home directory
/	the root (top level) directory

Text Editors

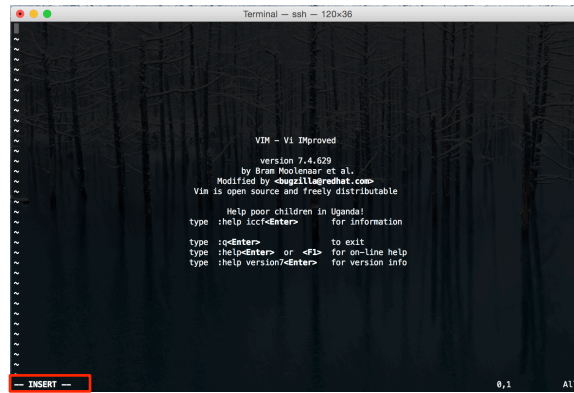
- Text editing
 - vi: standard Linux editor
 - vim: vi-iMproved
 - pico: easy-to-use text editor
 - emacs: powerful editor
 - many more...
 - No recommendation, pick one you are comfortable with.

Basic VI Commands

- Use vi to edit a file
 - vi filename
- Two modes: insert mode and display mode
 - i enter the insert mode
 - <esc> escape from the insert mode

VI Modes

- Insert Mode
 - You can modify the content of the file in the insert mode.



A terminal window titled "Terminal - ssh - 120x36" showing the Vim editor in Insert Mode. The screen displays the Vim startup screen with the following text:

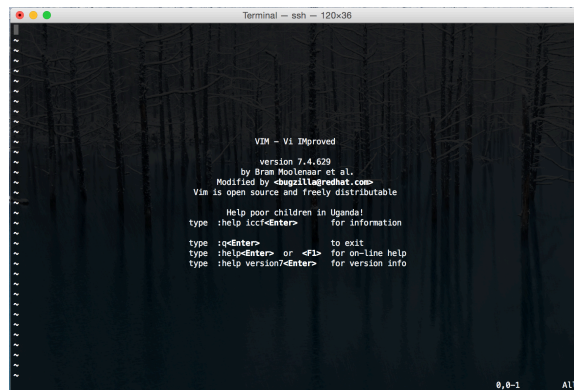
```
VIM - VI Improved
version 7.4.629
  by Bram Moolenaar et al.
  Modified by <aguzillagredat.com>
  Vim is open source and freely distributable

  Help poor children in Uganda!
  type :help iccf<Enter>    for information
  type :<Enter>            to exit
  type :help<Enter> or <F1> for on-line help
  type :help version?<Enter> for version info
```

At the bottom left, the text "-- INSERT --" is displayed in a red box. At the bottom right, the text "0,1 All" is visible.

VI Modes

- Display Mode
 - You can type in any VI commands in the display mode.



A terminal window titled "Terminal - ssh - 120x36" showing the Vim editor in Display Mode. The screen displays the Vim startup screen with the following text:

```
VIM - VI Improved
version 7.4.629
  by Bram Moolenaar et al.
  Modified by <aguzillagredat.com>
  Vim is open source and freely distributable

  Help poor children in Uganda!
  type :help iccf<Enter>    for information
  type :<Enter>            to exit
  type :help<Enter> or <F1> for on-line help
  type :help version?<Enter> for version info
```

At the bottom right, the text "0,0-1 All" is visible.

Basic VI Commands

- Saving and Quitting Commands:
 - `:w` Write/save the file
 - `:w file` Save to `file`
 - `:wq` Save the file and quit from vi
 - `:q!` Quit from vi, and discard all unsaved content
- Cursor Moving Commands:
 - `h/j/k/l` Move left/down/up/right
 - `:line` Go to Line Number `line`
 - `CTRL+F` Move to the next page
 - `CTRL+B` Move to the previous page

Basic VI Commands

- Cut (copy) and paste
 - Position the cursor where you want to start cutting /copying
 - Press `v` to select characters you want to cut/copy
 - Move the cursor to the end of what you want to cut /copy
 - Press `d/y` to cut/copy
 - Move to where you would like to paste
 - Press `P` to paste before the cursor, or `p` to paste after
- Shortcuts
 - `dd` Delete the current line
 - `yy` Copy the current line

Linux Shell Practice

- Login to your Linux account
- Print the current working directory
- List all files in the current working directory
- Create a new directory named EECS22
- Change into the new directory
- Create a text file named helloworld
- Insert "Hello World!" into the new file
- Copy the string and paste it on the second line
- Save and exit the new file
- Copy the new file to the upper directory
- Change to the upper directory
- Remove the original and copied files

Linux Shell Practice

- Login to your Linux account
- Print the current working directory `pwd`
- List all files in the current working directory `ls`
- Create a new directory named EECS22 `mkdir EECS22`
- Change into the new directory `cd EECS22`
- Create a text file named helloworld `vi helloworld`
- Insert "Hello World!" into the new file
- Copy the string and paste it on the second line
- Save and exit the new file `:wq`
- Copy the new file to the upper directory `cp helloworld ../`
- Change to the upper directory `cd ..`
- Remove the original and copied files
 - `rm helloworld`
 - `rm EECS22/helloworld`

Linux and VI Tutorials

- Linux tutorials
 - <http://linux.org.mt/article/terminal>
 - <http://www.linux-tutorial.info/modules.php?name=MContent&pageid=49>
- VI Tutorial
 - <http://www.ece.uci.edu/~chou/vi.html>

Programming Discussion

- Part 2 of Assignment 1
 - Collatz conjecture
 - https://en.wikipedia.org/wiki/Collatz_conjecture
 - Take any positive integer n .
 - If n is even, divide it by 2 to get $n / 2$.
 - If n is odd, multiply it by 3 and add 1 to obtain $3n + 1$.
 - Repeat the process indefinitely till you reach 1.
 - No mathematical proof available
 - Write a C program to calculate the Collatz sequence for any positive number input by the user.

Program Interface

- Demo:

```

=====
Collatz Conjecture Test:
=====

Enter a positive integer (or 0 to quit): 6
Collatz sequence: 6, 3, 10, 5, 16, 8, 4, 2, 1
Collatz conjecture is true for n=6.
Collatz sequence stops after 8 steps.
Maximum value is 16 at step 4.

Enter a positive integer (or 0 to quit): 7
Collatz sequence: 7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8,
4, 2, 1
Collatz conjecture is true for n=7.
Collatz sequence stops after 16 steps.
Maximum value is 52 at step 5.

Enter a positive integer (or 0 to quit): 0
Quit.

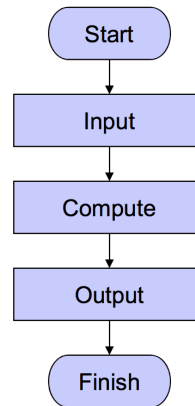
```

Assignment Discussion

- Collatz sequence
 - Keep calculating the Collatz sequence until the user inputs 0
 - Use the modulo (%) operator to determine an integer is even or odd
 - We have a maximum number of steps equal to 1 million.
 - Stopping after 1 million steps.
 - Collatz conjecture not clear for n=XXX.
 - Break the loop when we reach 1

General Program Structure

- General Program Structure
 - Input
 - Computation
 - Output
- Input: read input data
 - `scanf()`
- Computation
- Output: write output data
 - `printf()`



Bonus Part (5 Points)

- Print the maximum value reached in the sequence, together with its corresponding step number.
Collatz conjecture is true for $n=27$.
Collatz sequence stops after 111 steps.
Maximum value is 9232 at step 77.
- Only print the maximum value when we reach 1 before using up the maximum number of steps.
- Use the same files to submit the bonus part.

Assignment Setup

- Create a directory named `hw1` for Assignment 1
- Use your editor to create a C file named `collatz.c`
- Add your name and exercise number as a comment at the top of your file
- **Please use proper indentation in your source codes!**

Assignment Setup

- Compile your program
 - `gcc collatz.c -ansi -Wall -o collatz`
 - `./collatz`
- Submit the homework
 - Go to the parent directory of `hw1`
 - Submission command:
 - `turnin22`
 - Change the permission of the directory and files
 - `chmod -R 755 hw1`

Submission

- Please use the *exact* file names:
 - `collatz.c`
 - `collatz.txt`
 - `collatz.script`
- Source file: `collatz.c`
 - The complete program that is compilable
- Text file: `collatz.txt`
 - Briefly describe what your program does and why you implemented it in this way

Typescript

- A text file that captures an interactive session within the Linux shell.
- The typescript shows that your program runs correctly with the given input.
- Start the typescript with a command `script`, and stop it with `exit` or `ctrl-D`.
 - Don't start a text editor when recording.
 - Rename `typescript` after recording.

HelloWorld.script

```
zuma% script
Script started, file is typescript
zuma% gcc HelloWorld.c -ansi -Wall -o HelloWorld
zuma% ./HelloWorld
Hello World!
zuma% exit
exit
Script done, file is typescript
zuma% mv typescript HelloWorld.script
zuma% more HelloWorld.script
Script started on Wed Jun 24 14:52:39 2015
zuma% gcc HelloWorld.c -ansi -Wall -o HelloWorld
zuma% ./HelloWorld
Hello World!
zuma% exit
exit
Script done on Sun Sep 27 14:52:54 2015
```

Submission

- Assignment 1 due:
 - Thursday October 6, 2016 at 6:00 pm
- You can use the same command to update your submitted files until the submission deadline.