CSE 230
Intermediate Programming in C and C++
Code in UNIX System

Fall 2017

Stony Brook University

Instructor: Shebuti Rayana

Write, Compile, Execute your First C Code in UNIX Environment

- Instructions for your own Linux System
  http://www3.cs.stonybrook.edu/~cse230/CSE230_1.html

- Instructions for Sparky
  http://www3.cs.stonybrook.edu/~cse230/CSE230_2.html

For Sparky

- Once you have an account on Sparky, log in using ssh

- Use the following command
  `ssh netid@sparky.ic.sunysb.edu`

- By default this command will connect you to port 22.

- This command will prompt for your password. Give your password.

- Username = netid and password = netid password

- Your Web Space is 20MB in size

- Your directory on Sparky will be: /usr/www/Stu/netid

- Use `pwd` command to print the name of the current directory to see your location

Where to write the commands?

- ■ Windows User
- – SSH Secure Shell
- ■ available on Softweb or on any SINC Windows machine. Goto https://softweb.cc.stonybrook.edu/?ssh# and download

- ■ https://it.stonybrook.edu/sites/it.stonybrook.edu/files/docs/kb/Sparky-Webspace-Windows-Users.pdf

- ■ Mac User
- – Terminal
- ■ In spotlight search, write terminal and enter. Terminal comes with mac os.

File Transfer

- **Windows User**

- SSH Secure File Transfer

- SSH client includes both SSH shell and SSH file transfer

- Instructions to install and use: https://it.stonybrook.edu/sites/it.stonybrook.edu/files/docs/kb/How-to-use-SSH-File-Transfer.pdf

- **Mac User**

- Fetch

- Download link: https://it.stonybrook.edu/software/title/fetch

- Instructions to install and use: https://it.stonybrook.edu/sites/it.stonybrook.edu/files/docs/kb/Sparky-WebSpace-Mac-Users.pdf

Instructions Link

- https://it.stonybrook.edu/help/kb/sparky-web-space

How to access Sparky for Windows users using SSH (PDF)

How_to_use_SSH_File_Transfer (PDF) How to access Sparky for Mac users using Terminal and Fetch (PDF)

How to access the Sparky UNIX server (PC or Mac) (PDF) Try Cyberduck on your Personal Computer to Access Sparky!(PDF)

Creating Folder

- Your directory in Sparky: /usr/www/Stu/netid

- Now if you want to create separate directory for each homework use `mkdir`

  - `mkdir hw1`

  - This will create a `hw1` folder in /usr/www/Stu/netid

  - Getting inside hw1: `cd hw1`

  - Getting out of hw1: `cd ..`

  - Checking list of files and folders: `ls`

Important VIM commands

- http://www.radford.edu/~mhtay/CPSC120/VIM_Editor_Commands.htm

- https://vim.rtorr.com/

Your First Code

```
1   #incluse <stdio.h>
2
3   int main(void)
4   {
5       /*print hello world*/
6       printf("Hello World!!!");
7       return 0;
8   }
```

Create a practice folder and getting inside:
```
mkdir practice
cd practice
```

Writing your First Code

- **Opening a New File**

- Step 1 type **`vim`** ***`filename`*** (create a file named *filename e.g. hello.c*)

- Step 2 type **`i`** (switch to insert mode)

- Step 3 enter text (enter your program)

- Step 4 hit **`Esc key`** (switch back to command mode)

- Step 5 type **`:wq`** (write file and exit vim)

Compile and Execute

- Use `gcc` command to compile and build the program.

- Compile using gcc:
- `gcc hello.c`   [output goes to a.out]
- `gcc -o hello hello.c`

- Execute using gcc:
- `./a`
- `./hello`

Makefile Tutorial

- http://www.cs.colby.edu/maxwell/courses/tutorials/maketutor/

- Writing your own simple `makefile` so you do not have to re-write the command multiple times you want to execute your code

- Create a file named `makefile` and write the following lines in it
  ```
  hello: hello.c
  gcc -o hello hello.c
  ```

- Close the file

- Run: `make`

- Write `./hello` to see the output

Editing Existing Code

- Step 1 type **`vim filename`** (edit the existing file named filename)

- Step 2 move around the file using **`h`**/**`j`**/**`k`**/**`l`** key or any appropriate command
  - *`h` Moves the cursor one character to the left*
  - *`l` Moves the cursor one character to the right*
  - *`k` Moves the cursor up one line*
  - *`j` Moves the cursor down one line*
  - *`nG` or `:n` Cursor goes to the specified (n) line (ex. 10G goes to line 10)*

- Step 3 edit required text (`r`eplace or `d`elete or `i`nsert)

- Step 4 hit **`Esc`** `key` (exit from insert mode if you insert or replace text)

- Step 5 type **`:wq`**

Deleting lines

- Delete all lines
  `:1,$d`

- Delete line 2
  `:2d`

- Delete lines 2-5
  `:2,5d`