

CS120 – Computer Science I

Lecture 2

Instructor: Jia Song

Putty



- wormulon.cs.uidaho.edu

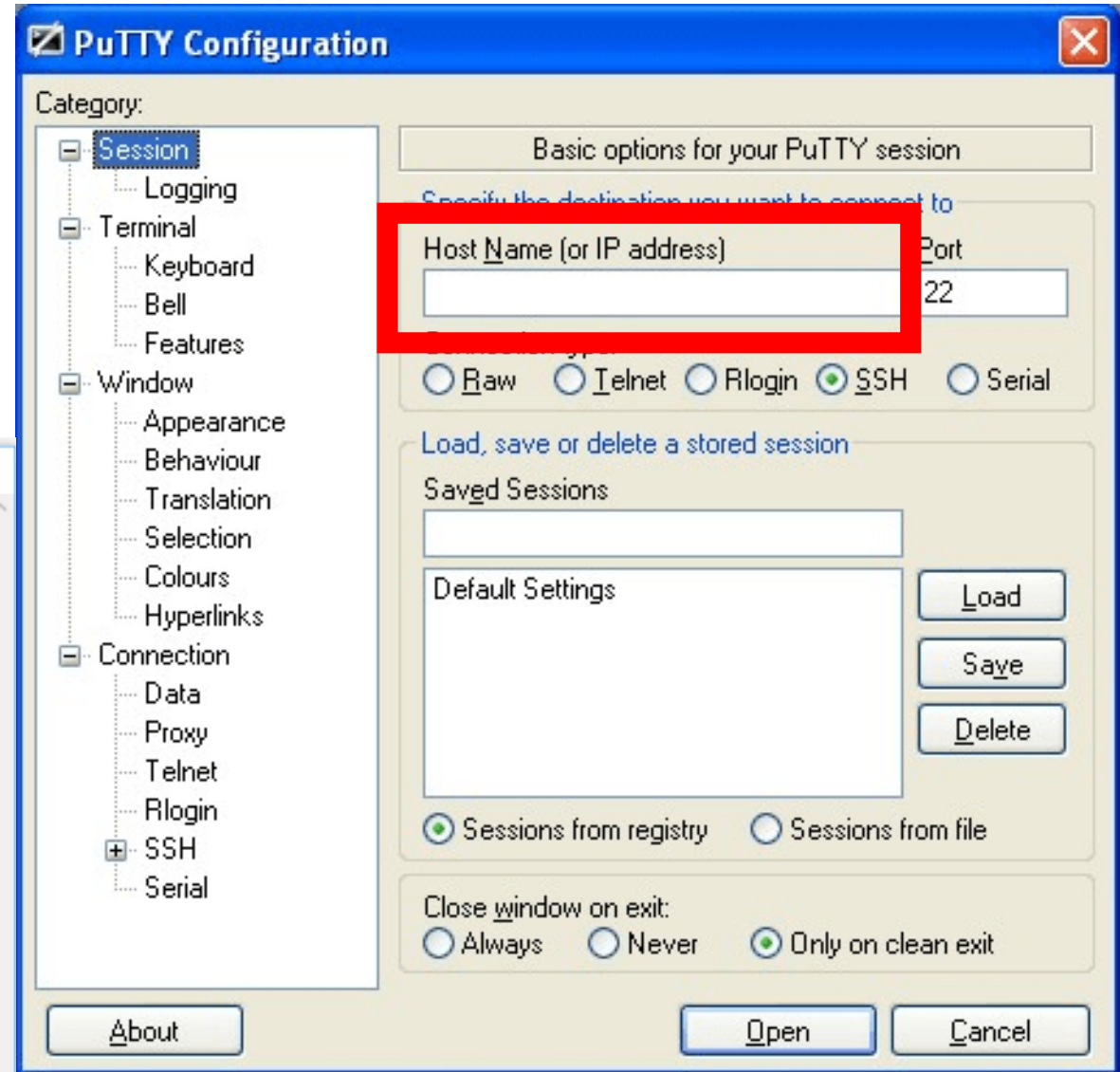
```
cs-wormulon.cs.uidaho.edu - PuTTY
login as: jsong

*****
* WARNING: To protect the system from unauthorized use and to *
* ensure that the system is functioning properly, activities *
* on this system are monitored recorded and subject to audit. *
* Use of this system is expressed consent to such monitoring *
* and recording. Any unauthorized access or use of this system *
* is prohibited and subject to criminal and civil penalties. *
*****

jsong@wormulon.cs.uidaho.edu's password:
Last login: Thu Jan 11 20:01:11 2018 from cpe-75-87-251-35.natnow.res.rr.com

FAQ: http://faq.cs.uidaho.edu
cshelp@uidaho.edu

* * *
-bash-4.1$
```



Transfer files from and to wormulon

- Windows machines:
- WinSCP (available through ITS)
- <http://www.uidaho.edu/infrastructure/its/services/software/its-tools>



- Mac machines:
- Cyberduck
- <https://cyberduck.io/?l=en>



Basic Unix Commands

- **ls** --- lists your files.
ls -l --- lists your files in 'long format', which contains lots of useful information.
- **cd *dirname*** --- change directory.
- **mkdir *dirname*** --- make a new directory.
- **pwd** --- tells you where you currently are.
- **more *filename*** --- shows the first part of a file, just as much as will fit on one screen. Just hit the space bar to see more or **q** to quit.
- **mv *filename1 filename2*** --- moves a file (i.e. gives it a different name, or moves it into a different directory)
- **cp *filename1 filename2*** --- copies a file
- **rm *filename*** --- removes a file.

Comments in C++

- Notes in program describing what code does
- Perform no action in program
- Two ways to comment in C++:
 - line comments using `//` (two slashes)
 - when `//` encountered, remainder of line ignored
 - works only on that line

```
//This is comment
```
 - Block comments `/* A Comment */`

```
/* This is a multi-line comment  
   This line is comment  
   This is the third line of comment  
*/
```

libraries

- A **library** is a separate file that contains additional code that the program uses.
- C++ has many built-in features.
 - They are not automatically included in program.
 - `#include <iostream>`
 - commands starting with a # sign are preprocessor directives.
- Where do the libraries located?
- `using namespace std;`
 - so the compiler knows the program is using the `standard (std)` namespace.
 - or `std::cout << "hello world!";`

The main function

- Every C++ program has a primary function that must be assigned the name `main`.
- `int main()`
 - The `int` indicates that `main` return an integer value to the operating system.
 - In standard C++, value 0 is returned upon successful execution of `main`.

```
{ <---- opening brace  
/*function body*/  
} <---- closing brace
```

C++ Input/Output

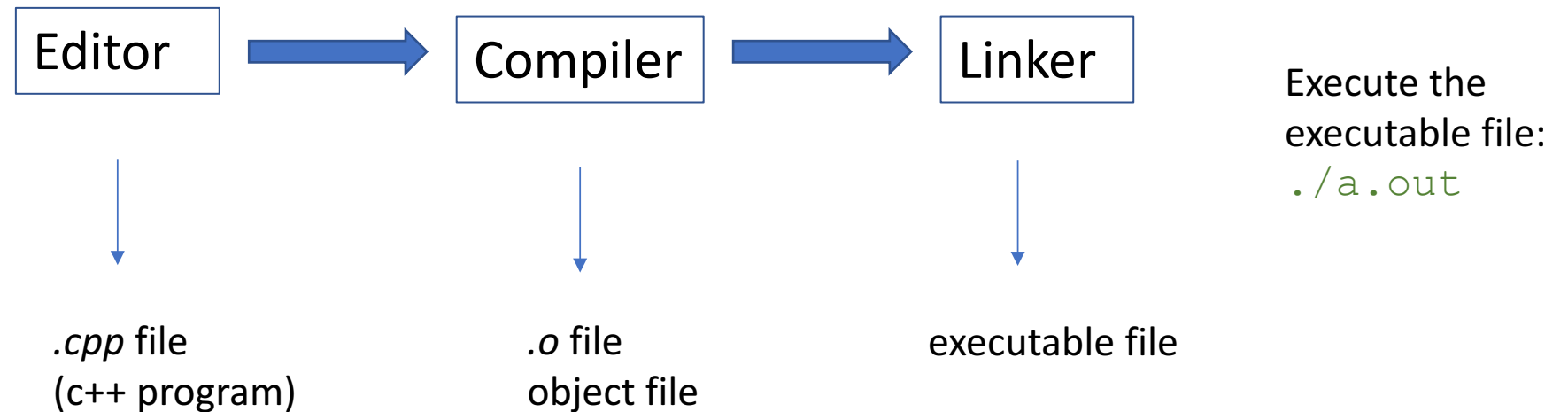
- `cout` (output stream) and `<<` (insertion operator) print the characters enclosed in double quotes to the screen.
- `cout << "Hello World!";`

C++ Syntax

- Rules for writing statements
 - Semicolon serve as statement terminator
 - Case sensitivity
 - cout is different from COUT, Cout, or cOut
 - Blank spaces
 - int ma in() (wrong)
 - c out (wrong)
 - Spacing

C++ Environment

- Because computers cannot directly execute a program written in a high-level language... --
Compiler & interpreter



- `g++ filename` (to compile the c++ program)
- `g++ -c filename` (get the .o file)
- `g++ -o newname filename` (to give the executable file a name, the default is a.out)
- `g++ --help` (show more options of g++)

Creating new lines in output

- two easy ways:
- Creating a new line with `\n` (backslash and n, no space between them)
 - `cout << "hello world!\n";`
 - `cout << "\nhello world!";`
 - There are many character escape sequences. e.g. `\0`, `\t`, `\\`
- Creating a new line with `endl`
 - `cout << "hello world!" << endl;`
 - `cout << endl;`
 - `cout << endl << endl << endl;`

Next week

- Monday Jan 15th - no classes
- Labs on Tuesday (sections 1 and 2) and Thursday (section 3) - JEB321
- Lectures will cover chapter 2 (variables, mathematical expression, `if`)
- Assignments:
 - Finish assignment #0 before lab
 - Reading assignment - Read chapter 1 and 2