



CS1101C

Briefing: UNIX and C Programming

Zhou Lifeng

zhoulife@comp.nus.edu.sg

23 January 2010



Start Off

- First of all, you need a SoC account to begin your programming journey (login to Sunfire). Apply for your SoC account at <https://mysoc.nus.edu.sg/~newacct> and follow instructions step by step.
- You will also need a CourseMarker account for submission of take-home labs, apply at <https://mysoc.nus.edu.sg/~cmarker/get-password.cgi>
- You'd better get above two accounts ready by week 3 so that you can focus on course materials.

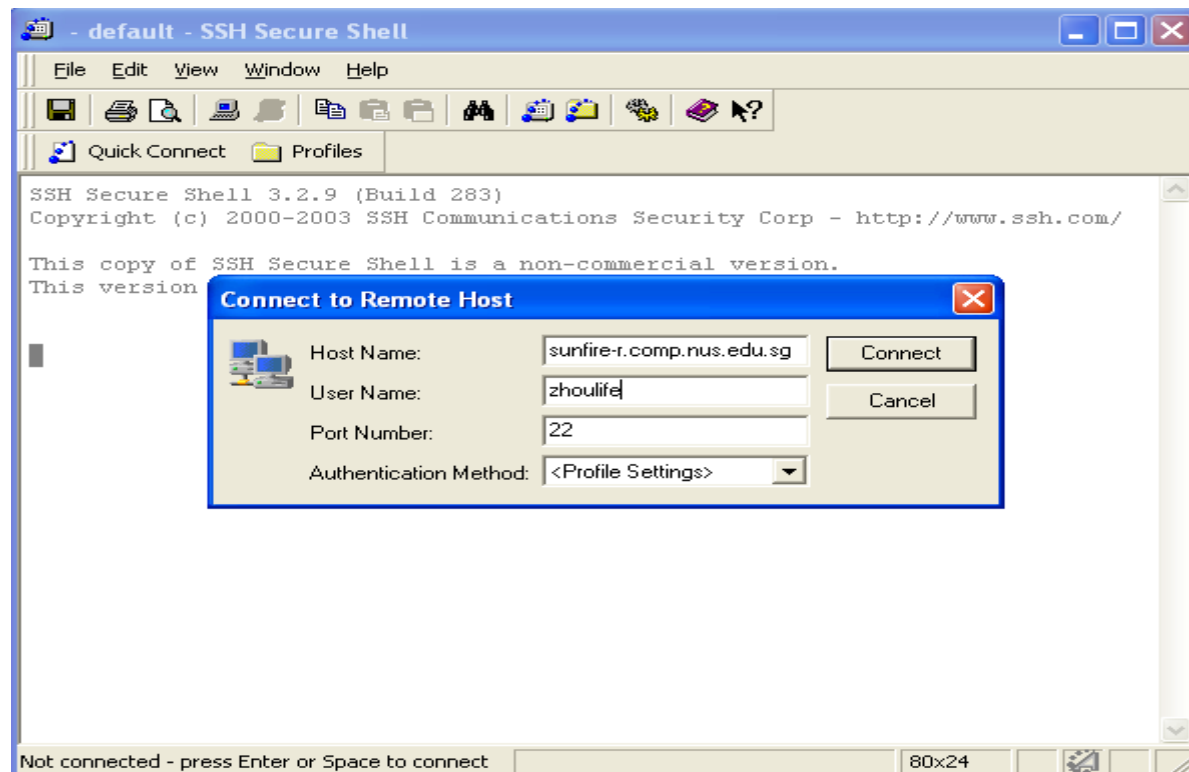


Sunfire Server

- Sunfire is a UNIX server located in the SoC Machine Room. It is the sole platform that we will carry out C programming (sit-in labs) in CS1101C.
- To connect to Sunfire, you need to install a software called “SSH Secure Shell” which is available in IVLE workbin.
- All computers in SoC PL labs have already pre-installed “SSH Secure Shell”.

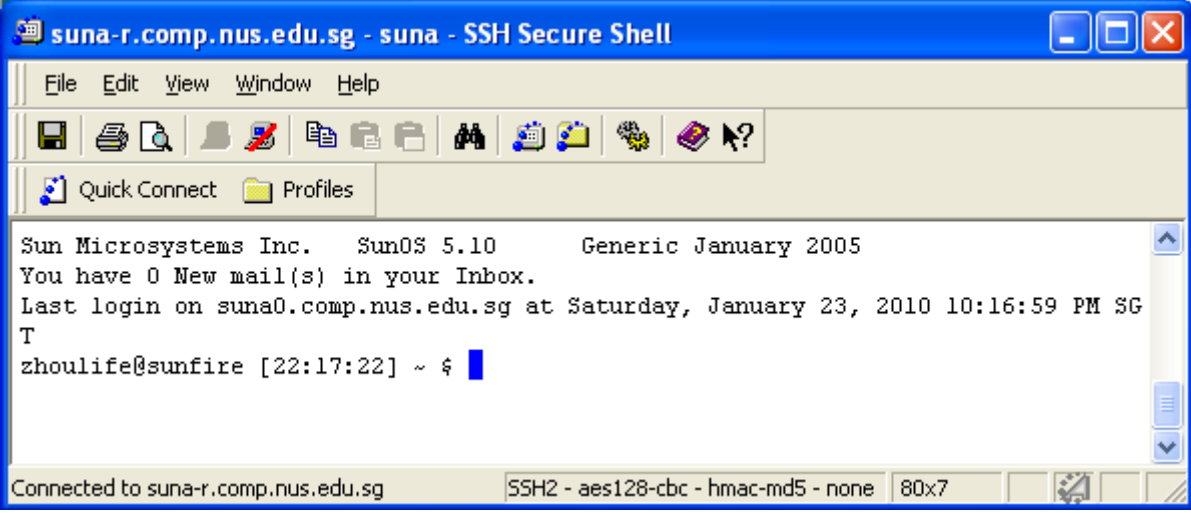
Connect to Sunfire from home

- Launch a “SSH Secure Shell Client” (typically available on desktop after installation), click “Quick Connect”
- Type Sunfire name and your SoC UNIX account (details on next page).



Connect to Sunfire from home (2)

- If you connect to Sunfire on campus, enter host name as: sunfire.comp.nus.edu.sg; if you connect to Sunfire off campus (e.g., at home), enter host name as: sunfire-r.comp.nus.edu.sg
- Enter user name as: [your SoC UNIX account](#)
- You will be prompted for password after you press “Enter” key; enter your UNIX account [password](#) and you are then connected to Sunfire.



The screenshot shows a window titled "suna-r.comp.nus.edu.sg - suna - SSH Secure Shell". The window has a menu bar with "File", "Edit", "View", "Window", and "Help". Below the menu bar is a toolbar with various icons. The main text area displays the following output:

```
Sun Microsystems Inc.  SunOS 5.10      Generic January 2005
You have 0 New mail(s) in your Inbox.
Last login on suna0.comp.nus.edu.sg at Saturday, January 23, 2010 10:16:59 PM SG
T
zhoulife@sunfire [22:17:22] ~ $
```

The status bar at the bottom of the window shows "Connected to suna-r.comp.nus.edu.sg", "SSH2 - aes128-cbc - hmac-md5 - none", and "80x7".

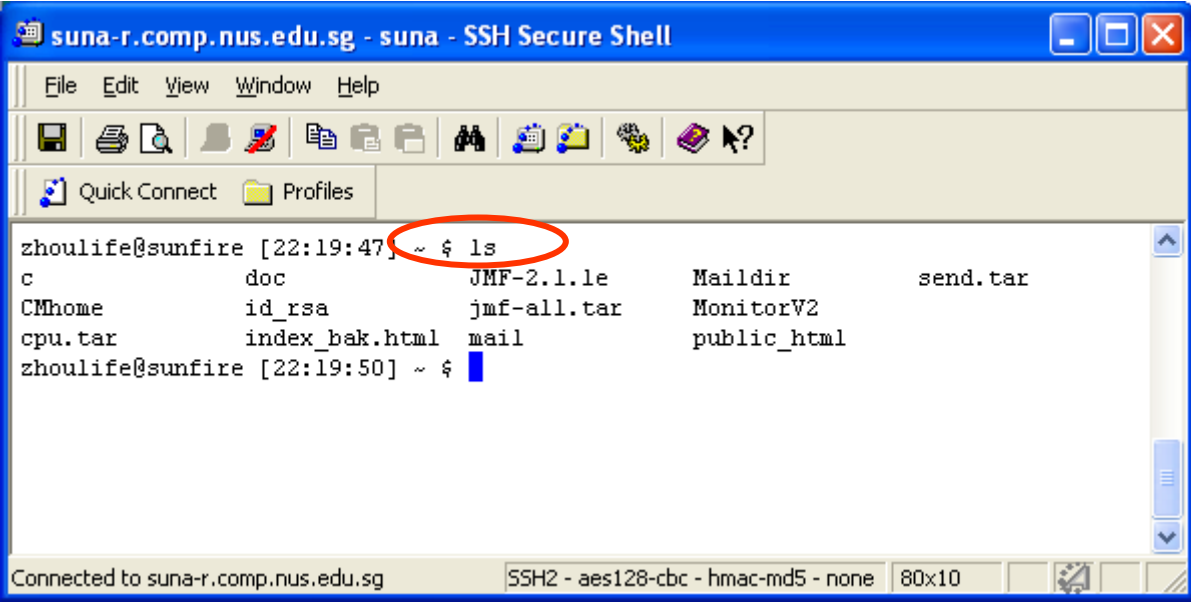


Basic Unix commands

- In a UNIX shell (like sunfire), you need a lot of typing but much less mouse clicking, compared with Windows operating system which you might be more familiar with.
- There are a few useful commands that you need to remember which will facilitate your navigation in the UNIX world.
- Practice is the best way to recognize UNIX commands. Gradually you will be more and more familiar with UNIX commands – so don't worry too much at the beginning.

ls command

- `ls` command (means list directory contents) will enable you to see all the files and subfolders in current directory.
- There are a few more complex usage of `ls`, but first of all, be familiar with the simplest one – just “`ls`”.



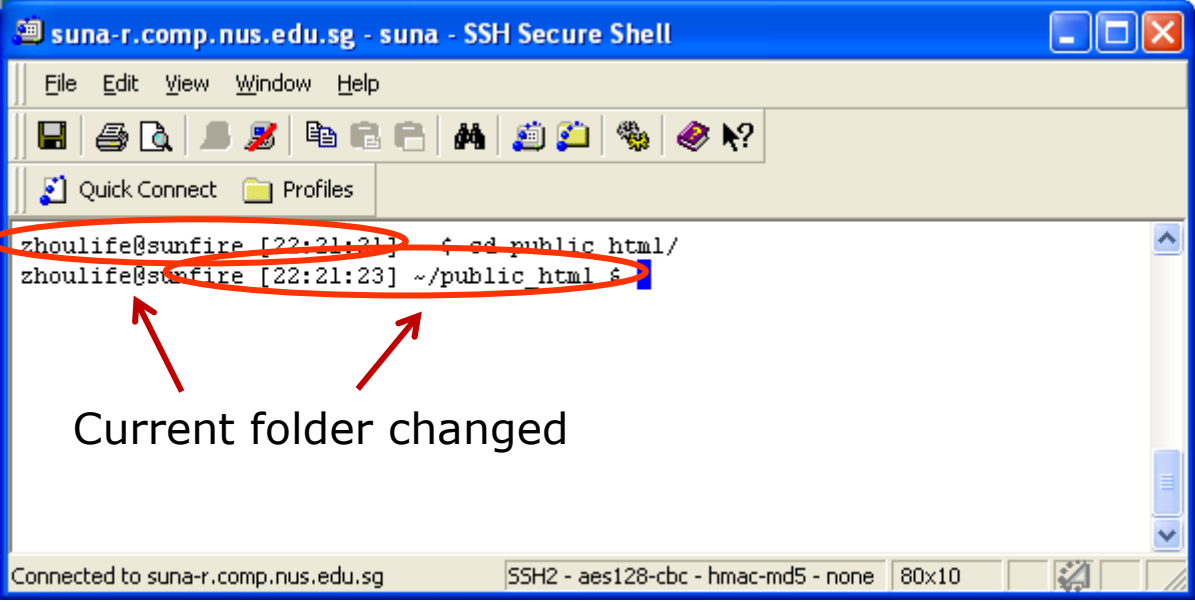
The screenshot shows a terminal window titled "suna-r.comp.nus.edu.sg - suna - SSH Secure Shell". The window has a menu bar (File, Edit, View, Window, Help) and a toolbar with various icons. Below the toolbar, there are tabs for "Quick Connect" and "Profiles". The main text area shows the following output:

```
zhoulife@sunfire [22:19:47] ~ $ ls
c          doc          JMF-2.1.1e      Maildir        send.tar
CMhome     id_rsa          jmf-all.tar     MonitorV2
cpu.tar    index_bak.html  mail            public_html
zhoulife@sunfire [22:19:50] ~ $
```

The command `ls` is circled in red. The output lists files and directories in a multi-column format. The status bar at the bottom indicates "Connected to suna-r.comp.nus.edu.sg", "SSH2 - aes128-cbc - hmac-md5 - none", and "80x10".

cd command (1)

- `cd` command allows you to enter a designated directory.
- e.g., `cd public_html`
- After that, your working directory will be changed to your desired one (see picture below).

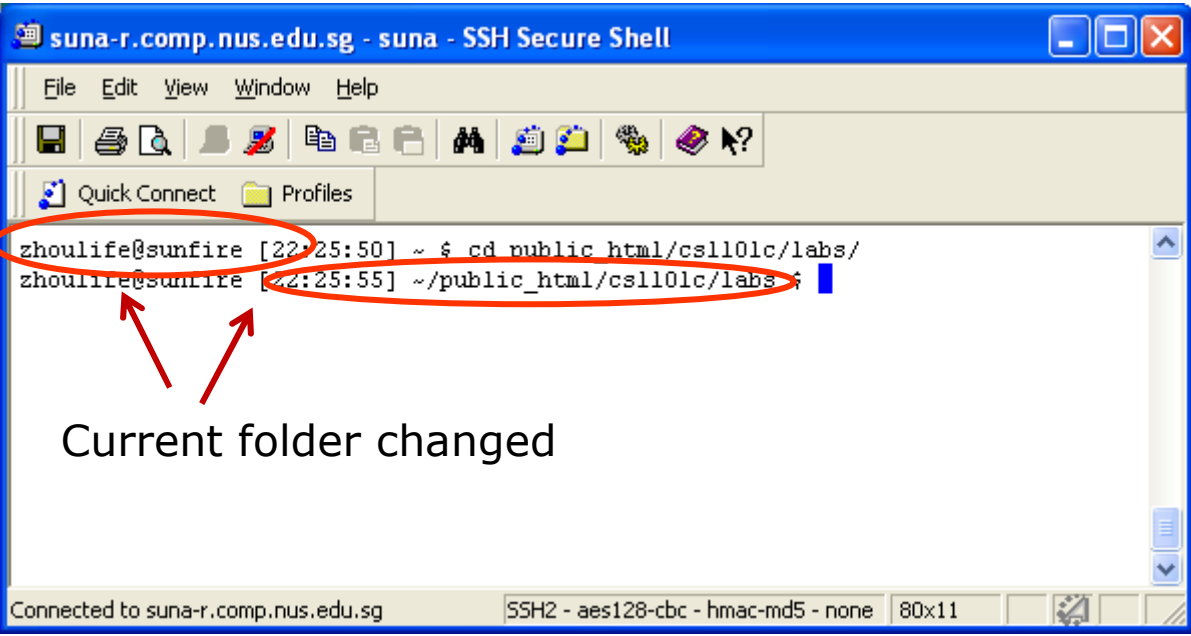


The screenshot shows a terminal window titled "suna-r.comp.nus.edu.sg - suna - SSH Secure Shell". The window has a menu bar (File, Edit, View, Window, Help) and a toolbar with various icons. Below the toolbar, there are tabs for "Quick Connect" and "Profiles". The terminal output shows two lines: "zhoulife@sunfire [22:21:21] \$ cd public_html/" and "zhoulife@sunfire [22:21:23] ~/public_html \$". A red oval highlights the first line, and two red arrows point from the text "Current folder changed" below to the first and second lines of the terminal output. The status bar at the bottom indicates "Connected to suna-r.comp.nus.edu.sg", "SSH2 - aes128-cbc - hmac-md5 - none", and "80x10".

```
suna-r.comp.nus.edu.sg - suna - SSH Secure Shell
File Edit View Window Help
[Icons]
Quick Connect Profiles
zhoulife@sunfire [22:21:21] $ cd public_html/
zhoulife@sunfire [22:21:23] ~/public_html $
Current folder changed
Connected to suna-r.comp.nus.edu.sg  SSH2 - aes128-cbc - hmac-md5 - none  80x10
```


cd command (2)

- You can also navigate to a subfolder of subfolder of subfolder... by giving full path. The folder structure of UNIX is similar to the Windows file system shown in a Windows explorer.
- e.g., `cd public_html/cs1101c/labs` (destination folder must exist)

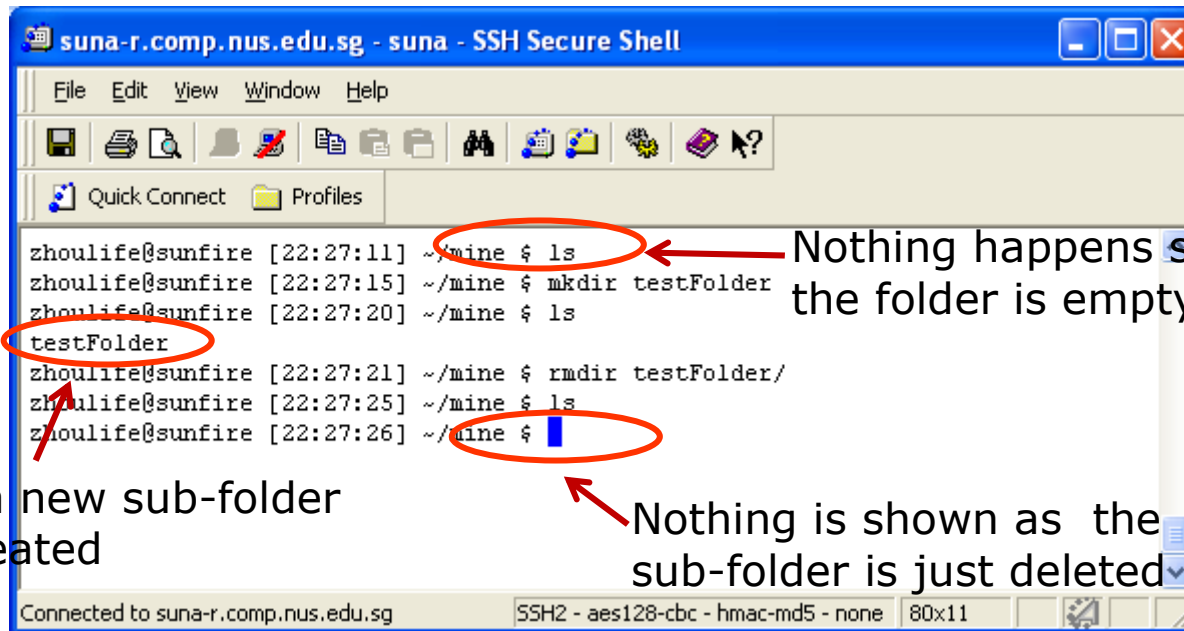


The screenshot shows a terminal window titled "suna-r.comp.nus.edu.sg - suna - SSH Secure Shell". The window has a menu bar (File, Edit, View, Window, Help) and a toolbar with various icons. Below the toolbar, there are tabs for "Quick Connect" and "Profiles". The terminal text shows the user "zhoulife@sunfire" at the prompt. The first line shows the command `cd public_html/cs1101c/labs/` being entered. The second line shows the prompt changed to `~/public_html/cs1101c/labs$`. Two red arrows point from the text "Current folder changed" below the terminal to the new prompt in the second line.

```
suna-r.comp.nus.edu.sg - suna - SSH Secure Shell
File Edit View Window Help
[Icons]
Quick Connect Profiles
zhoulife@sunfire [22:25:50] ~ $ cd public_html/cs1101c/labs/
zhoulife@sunfire [22:25:55] ~/public_html/cs1101c/labs$
Current folder changed
Connected to suna-r.comp.nus.edu.sg  SSH2 - aes128-cbc - hmac-md5 - none  80x11
```

mkdir and rmdir commands

- **mkdir** (means make directory) will create a sub-folder
- **rmdir** (means remove directory) will delete an *empty* sub-folder



The screenshot shows an SSH terminal window titled 'suna-r.comp.nus.edu.sg - suna - SSH Secure Shell'. The terminal displays the following commands and their outputs:

```
zhoulife@sunfire [22:27:11] ~/mine $ ls
zhoulife@sunfire [22:27:15] ~/mine $ mkdir testFolder
zhoulife@sunfire [22:27:20] ~/mine $ ls
testFolder
zhoulife@sunfire [22:27:21] ~/mine $ rmdir testFolder/
zhoulife@sunfire [22:27:25] ~/mine $ ls
zhoulife@sunfire [22:27:26] ~/mine $
```

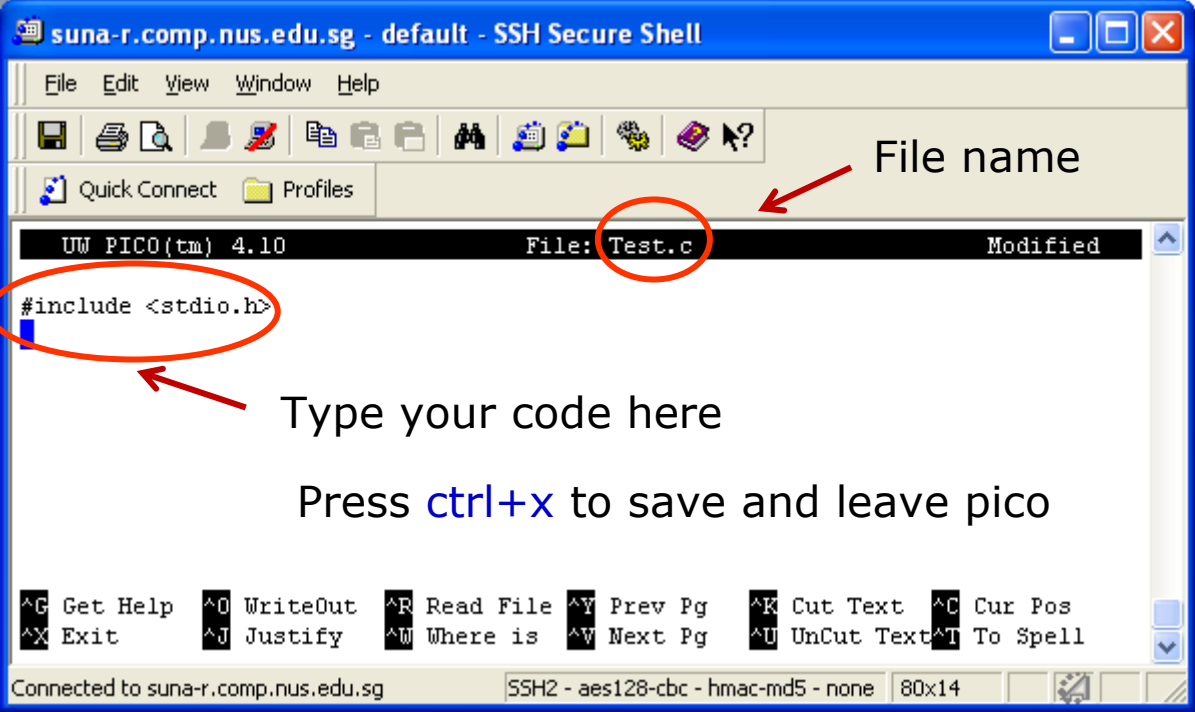
Red circles and arrows highlight specific parts of the terminal output:

- A red circle around the first `ls` command is annotated with the text: "Nothing happens since the folder is empty now".
- A red circle around the output `testFolder` is annotated with the text: "Show a new sub-folder just created".
- A red circle around the prompt after the `rmdir` command is annotated with the text: "Nothing is shown as the sub-folder is just deleted".

The status bar at the bottom of the window shows: "Connected to suna-r.comp.nus.edu.sg", "SSH2 - aes128-cbc - hmac-md5 - none", and "80x11".

Create a file

- You can either use “pico” or “vim” command to create a file or open an existing file.
- Basic usage: `pico filename` or `vim filename`
- E.g., `pico Test.c` (see picture below)



The screenshot shows a terminal window titled "suna-r.comp.nus.edu.sg - default - SSH Secure Shell". The PICO text editor is open, displaying a menu bar (File, Edit, View, Window, Help), a toolbar, and a status bar. The status bar shows "UW PICO(tm) 4.10", "File: Test.c" (circled in red with an arrow pointing to it labeled "File name"), and "Modified". The main editing area contains the text `#include <stdio.h>` (circled in red with an arrow pointing to it labeled "Type your code here"). Below the editing area, a message says "Press `ctrl+x` to save and leave pico". The status bar at the bottom lists various keyboard shortcuts: `^G` Get Help, `^O` WriteOut, `^R` Read File, `^Y` Prev Pg, `^K` Cut Text, `^C` Cur Pos, `^X` Exit, `^J` Justify, `^W` Where is, `^V` Next Pg, `^U` UnCut Text, `^T` To Spell. The bottom status bar indicates "Connected to suna-r.comp.nus.edu.sg", "SSH2 - aes128-cbc - hmac-md5 - none", and "80x14".

```
UW PICO(tm) 4.10      File: Test.c      Modified
#include <stdio.h>
```

Type your code here

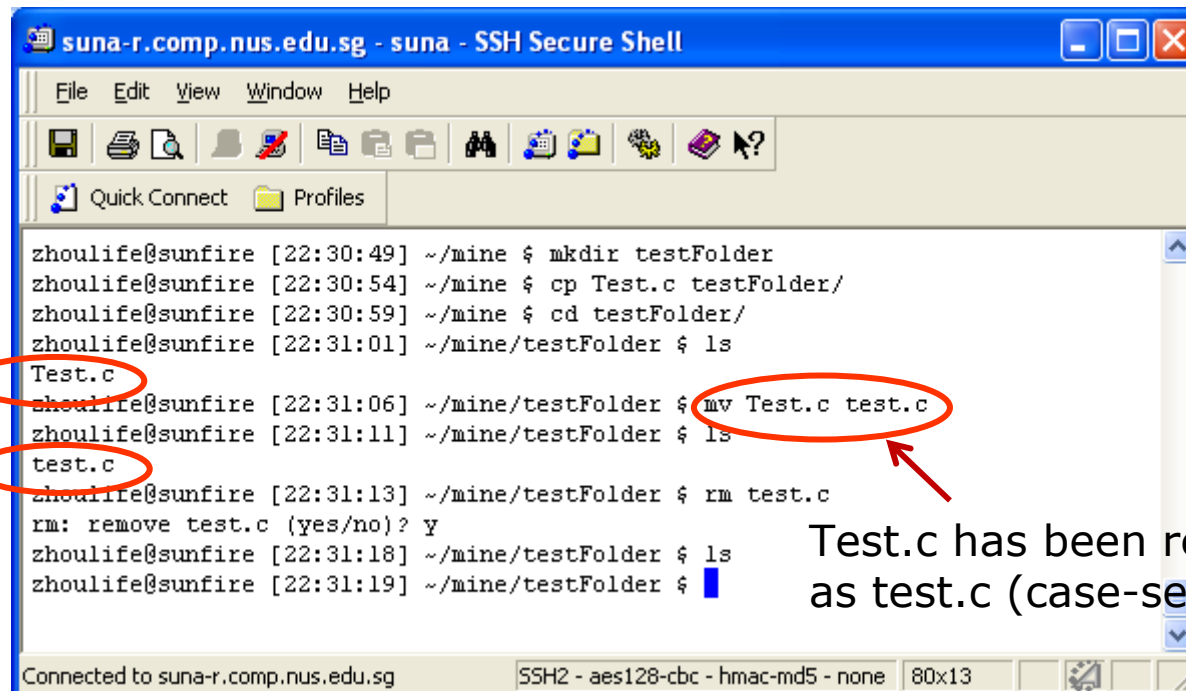
Press `ctrl+x` to save and leave pico

`^G` Get Help `^O` WriteOut `^R` Read File `^Y` Prev Pg `^K` Cut Text `^C` Cur Pos
`^X` Exit `^J` Justify `^W` Where is `^V` Next Pg `^U` UnCut Text `^T` To Spell

Connected to suna-r.comp.nus.edu.sg SSH2 - aes128-cbc - hmac-md5 - none 80x14

cp, mv and rm commands

- **cp** command copies a file to another folder.
- **mv** command move a file to another folder.
- **mv** command is also used to rename a file.
- **rm** command deletes a file.



```
suna-r.comp.nus.edu.sg - suna - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

zhoulife@sunfire [22:30:49] ~/mine $ mkdir testFolder
zhoulife@sunfire [22:30:54] ~/mine $ cp Test.c testFolder/
zhoulife@sunfire [22:30:59] ~/mine $ cd testFolder/
zhoulife@sunfire [22:31:01] ~/mine/testFolder $ ls
Test.c
zhoulife@sunfire [22:31:06] ~/mine/testFolder $ mv Test.c test.c
zhoulife@sunfire [22:31:11] ~/mine/testFolder $ ls
test.c
zhoulife@sunfire [22:31:13] ~/mine/testFolder $ rm test.c
rm: remove test.c (yes/no)? y
zhoulife@sunfire [22:31:18] ~/mine/testFolder $ ls
zhoulife@sunfire [22:31:19] ~/mine/testFolder $
```

Test.c has been renamed as test.c (case-sensitive)



C Programming on Sunfire

- The next few slides will show you two examples of simple C programming on Sunfire.
- First example will show you how to create a text file to store your program, compile it and run it.
- Second example will show you how to open an existing C program, modify it, save it, compile it and run it again.

Create and edit HelloWorld.c

- The picture below shows the editing interface of pico (by using command `pico HelloWorld.c`)
- All the code is typed in manually.
- Press `ctrl+x` to save your work and leave pico.

```
UW PICO(tm) 4.10 File: HelloWorld.c

#include <stdio.h>

int main(void)
{
    printf("Hello World\\n");
    return 0;
}
```

This is your expected program output

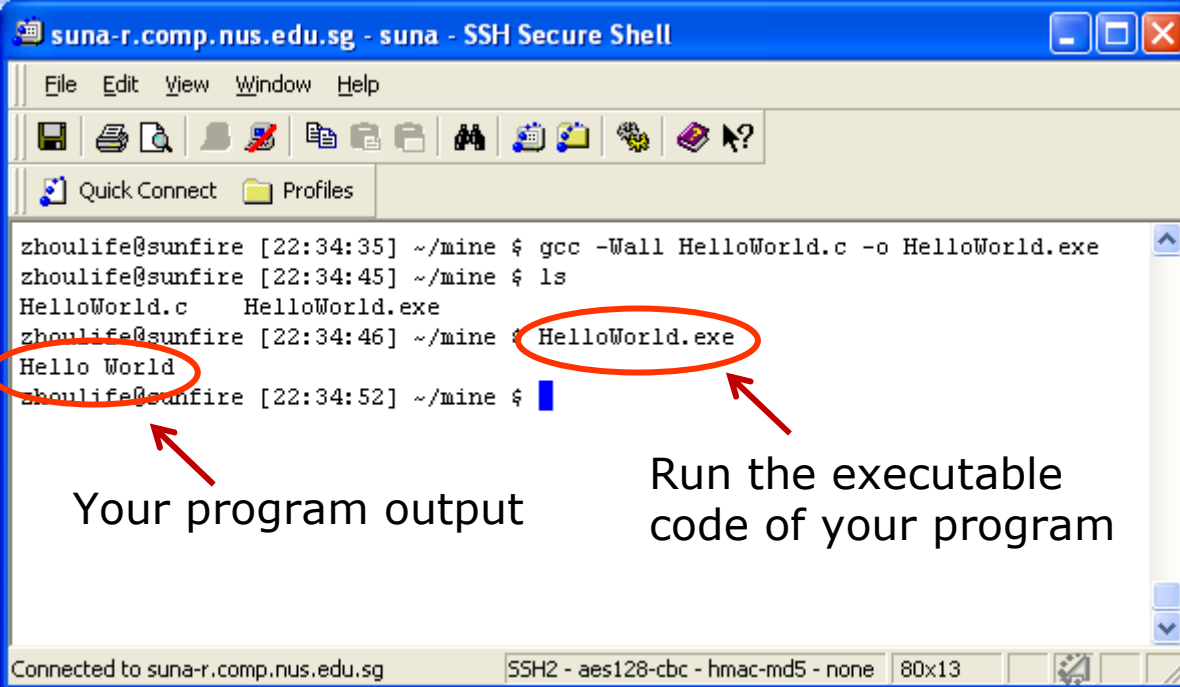
[Wrote 8 lines]

^G Get Help ^O WriteOut ^R Read File ^Y Prev Pg ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where is ^V Next Pg ^U UnCut Text ^T To Spell

Connected to suna-r.comp.nus.edu.sg SSH2 - aes128-cbc - hmac-md5 - none 80x13

Compile and run HelloWorld.c

- After returning from pico, type command `gcc -Wall HelloWorld.c -o HelloWorld.exe` to compile your code. You can find that a file called `HelloWorld.exe` is generated in the same folder.
- Type `HelloWorld.exe` and observe the output of your program.



The screenshot shows an SSH terminal window titled "suna-r.comp.nus.edu.sg - suna - SSH Secure Shell". The terminal displays the following commands and output:

```
zhoulife@sunfire [22:34:35] ~/mine $ gcc -Wall HelloWorld.c -o HelloWorld.exe
zhoulife@sunfire [22:34:45] ~/mine $ ls
HelloWorld.c  HelloWorld.exe
zhoulife@sunfire [22:34:46] ~/mine $ HelloWorld.exe
Hello World
zhoulife@sunfire [22:34:52] ~/mine $
```

Two red circles highlight the output "Hello World" and the command "HelloWorld.exe". Red arrows point from the text "Your program output" to the "Hello World" output and from "Run the executable code of your program" to the "HelloWorld.exe" command.

Modify HelloWorld.c

- Type `pico HelloWorld.c` again and modify your program. Change "Hello world" to "Hello Buddy".
- Save and exit pico (`ctrl+x`) , re-compile your code (`gcc -Wall HelloWorld.c -o HelloWorld.exe`) and run it again (`HelloWorld.exe`). You will get different output from last time!

```
UW PICO(tm) 4.10      File: HelloWorld.c

#include <stdio.h>

int main(void)
{
    printf("Hello Buddy\n");

    return 0;
}

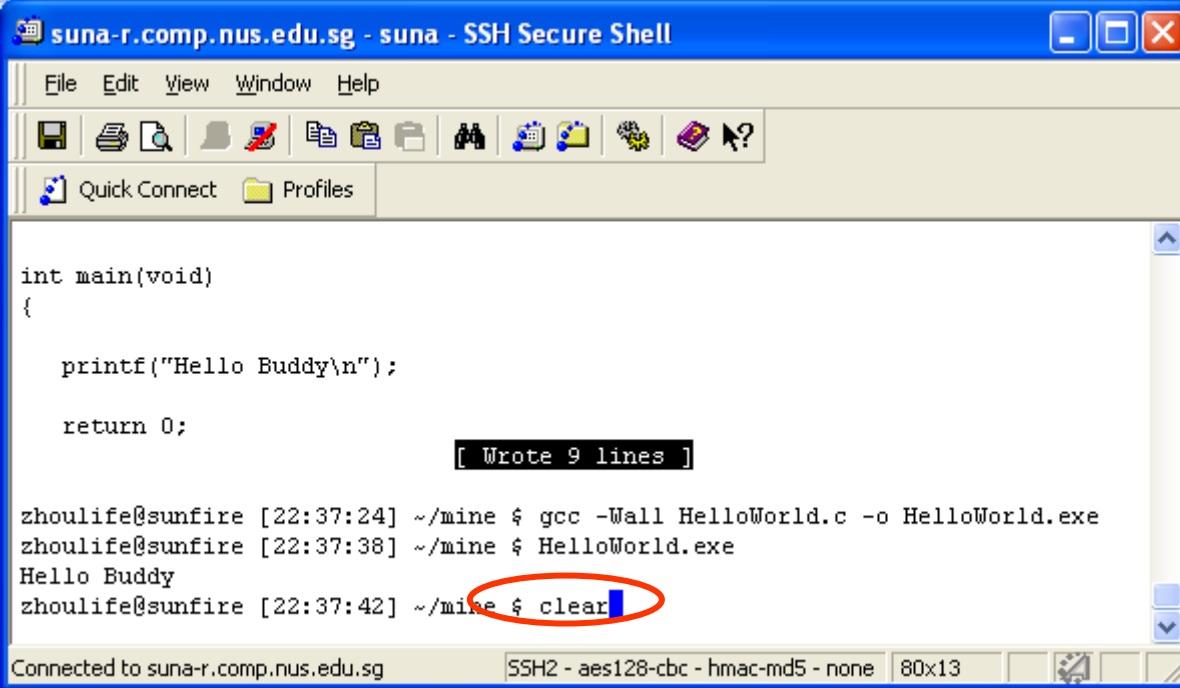
[ Wrote 8 lines ]

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Pg   ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where is   ^V Next Pg   ^U UnCut Text ^T To Spell

Connected to suna-r.comp.nus.edu.sg  SSH2 - aes128-cbc - hmac-md5 - none  80x14
```


clear command

- If the screen is too messy and you want to clear it, simple type command `clear`
- Try the command by yourself.



The screenshot shows a terminal window titled "suna-r.comp.nus.edu.sg - suna - SSH Secure Shell". The window contains a C program for "HelloWorld.c" and its execution. The program prints "Hello Buddy". The user then enters the command `clear`, which is circled in red. The terminal status bar at the bottom indicates the connection is to suna-r.comp.nus.edu.sg using SSH2 with aes128-cbc encryption, hmac-md5 authentication, and an 80x13 window size.

```
int main(void)
{
    printf("Hello Buddy\n");
    return 0;
}
[ Wrote 9 lines ]
zhoulife@sunfire [22:37:24] ~/mine $ gcc -Wall HelloWorld.c -o HelloWorld.exe
zhoulife@sunfire [22:37:38] ~/mine $ HelloWorld.exe
Hello Buddy
zhoulife@sunfire [22:37:42] ~/mine $ clear
```



Logout

- After finishing your job, you may want to exit from Sunfire.
- The command is quite simple: `exit`
- Try it and say “bye-bye” 😊



Congratulations!

- So far you have created and run your first C program on Sunfire, congratulations!
- Still got a lot not clear? Don't worry, more details will be introduced in week 3 lab.
- You will gain more experience after days and weeks.
- Two useful links on more completed Unix commands and programming:
<http://www.comp.nus.edu.sg/~cs1101c/gettingStarted.html>

http://www.comp.nus.edu.sg/~cs1101x/3_ca/labs/lab0/unix_intro.html

(please ignore those irrelevant parts if applicable)