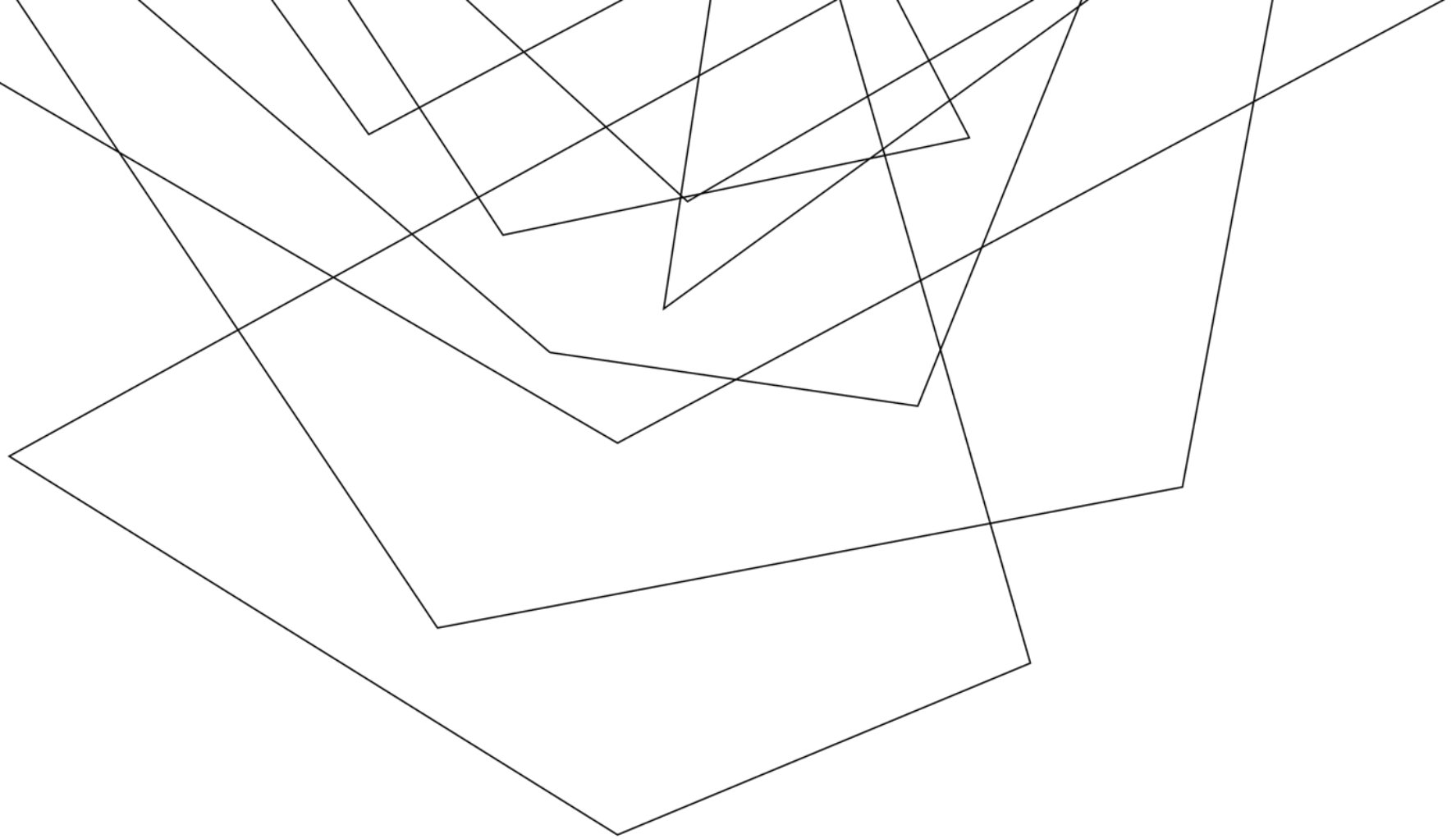


CMPT 125 D100 LAB1

TA

TOPICS FOR TODAY

- Basic commands in Linux
- VS Code configuration
- Hello World (VS Code)
- Hello World (Command Line)

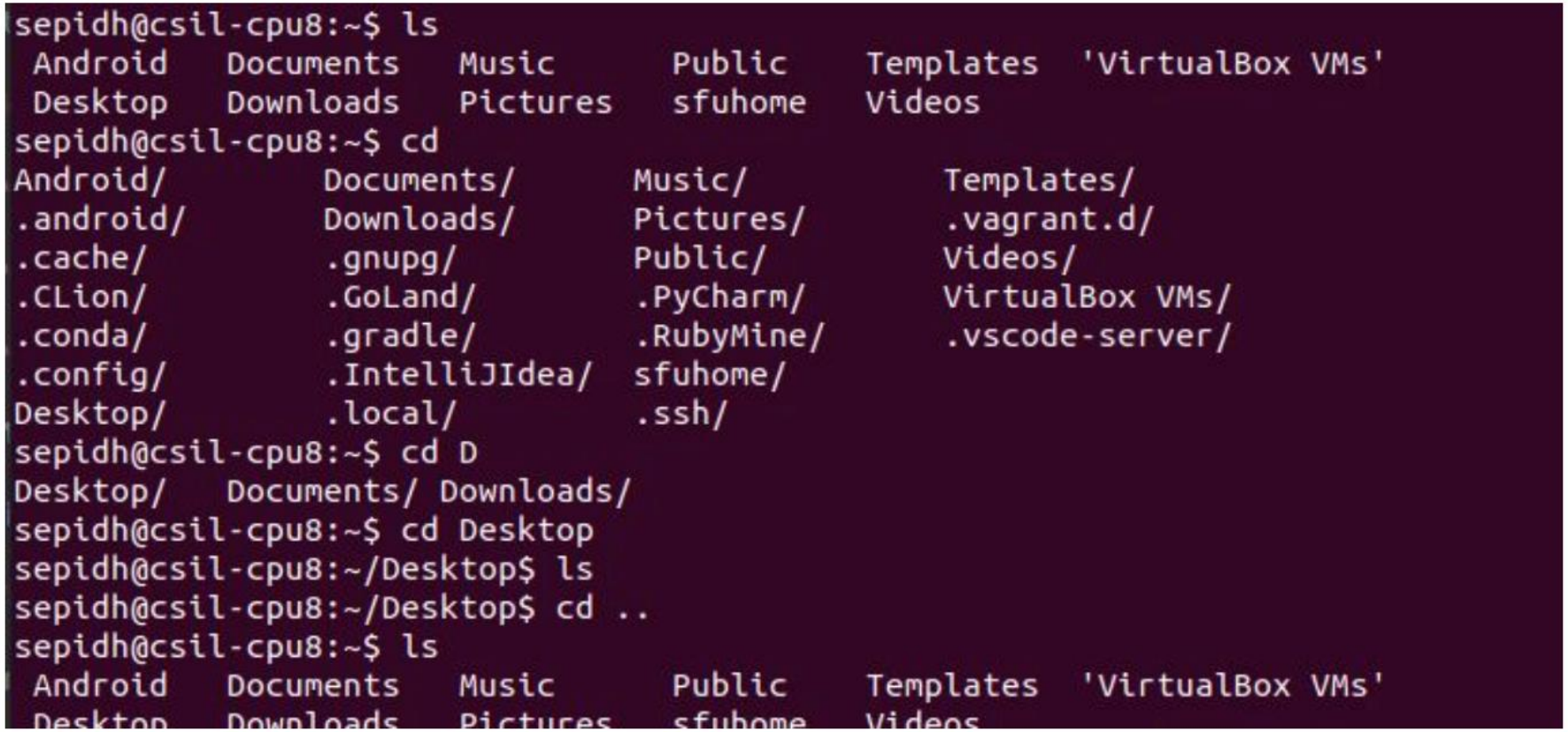
Basic commands in Linux

- pwd : will give you the address of directory you are in.
- ls : use it to show the files and directories that are in the directory you are in.

```
sepidh@csil-cpu8:~$ pwd
/home/sepidh
sepidh@csil-cpu8:~$ ls
 Android   Documents   Music      Public    Templates  'VirtualBox VMs'
 Desktop   Downloads   Pictures   sfuhome   Videos
sepidh@csil-cpu8:~$
```

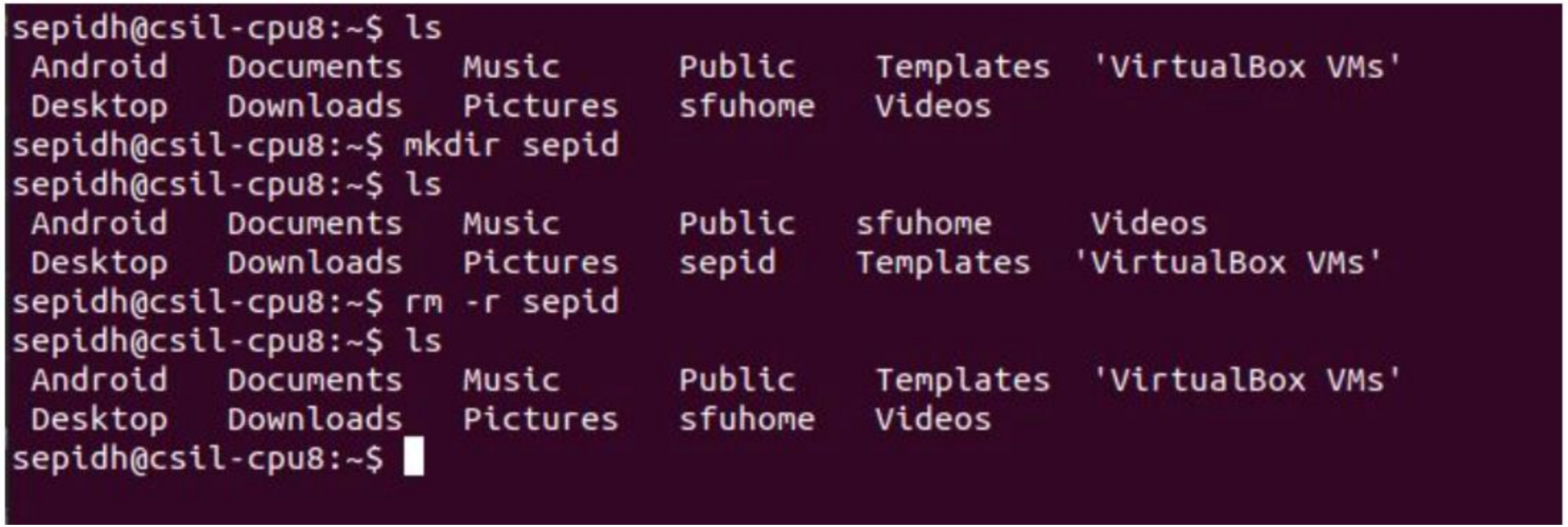
Basic commands in Linux

- cd : To navigate through the Linux directories and files. if you write cd and press tab it will show you possible directories you can enter using cd, you can write part of file name to make the search space for that smaller. (use cd .. to go backward.)



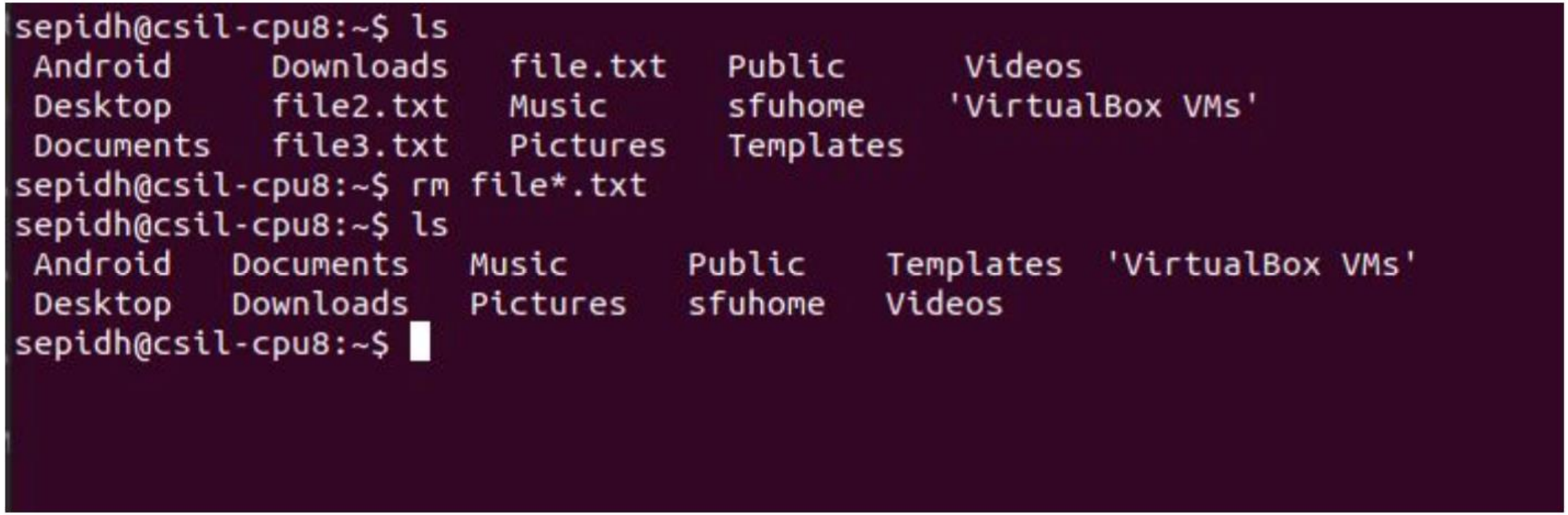
Basic commands in Linux

- mkdir : use this to create new directory
- rm : use these to remove files, and directories. (For directories you need to use rm -r)



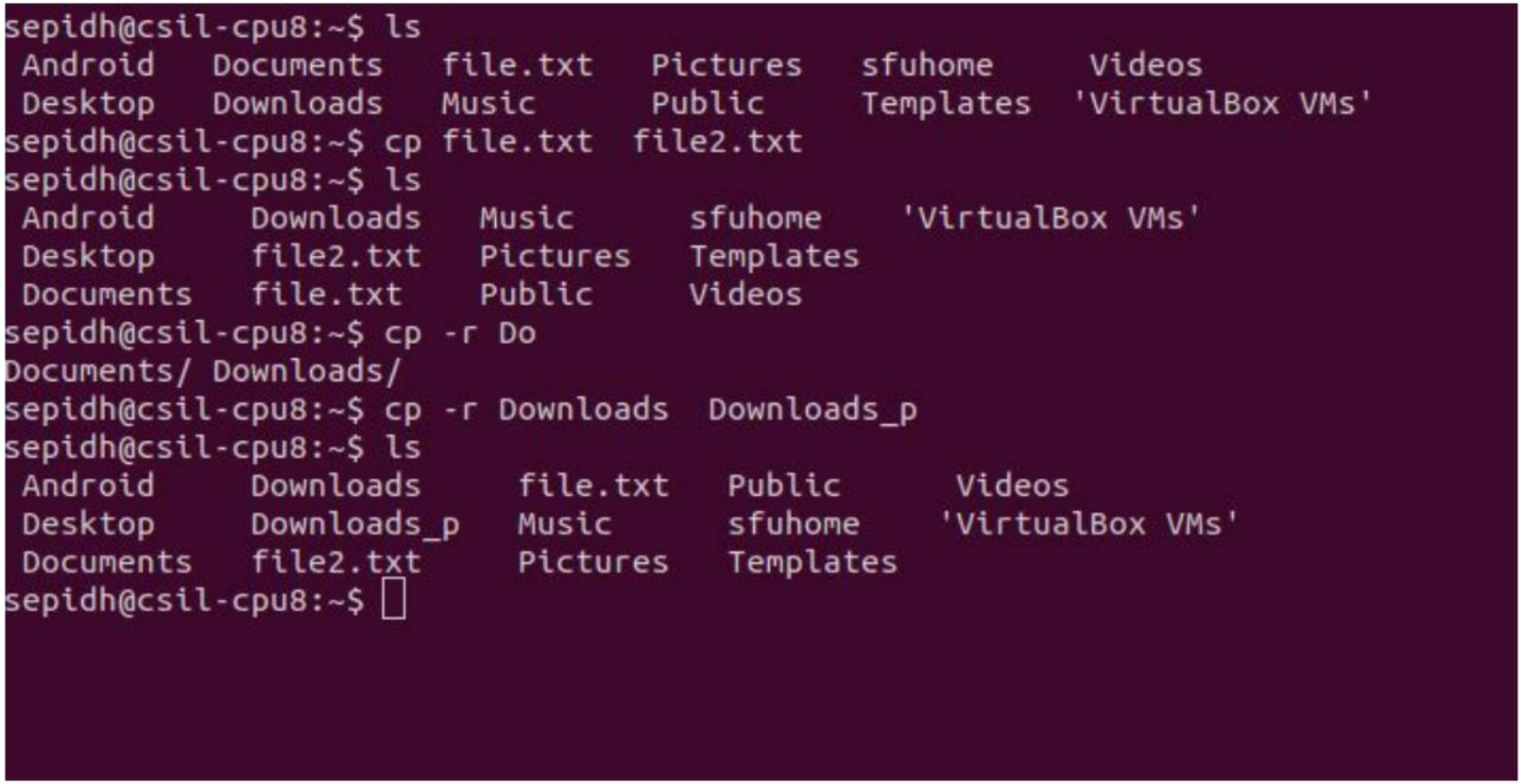
Basic commands in Linux

- *: means anything. for example if you say rm file*.txt you mean remove all file“something”.txt



Basic commands in Linux

- cp: use cp for copying file and directories, for directories again you need to do cp -r. Same as other command tab can give you possible options for being copied.



Basic commands in Linux

- mv: use mv to move files and directories.

```
sepidh@csil-cpu8:~$ ls
 Android     Downloads     file.txt   Public      Videos
 Desktop     Downloads_p   Music      sfuhome    'VirtualBox VMs'
 Documents   file2.txt     Pictures   Templates
sepidh@csil-cpu8:~$ mv file.txt Downloads/
sepidh@csil-cpu8:~$ ls
 Android     Downloads     Music      sfuhome    'VirtualBox VMs'
 Desktop     Downloads_p   Pictures   Templates
 Documents   file2.txt     Public     Videos
sepidh@csil-cpu8:~$ cd Downloads
sepidh@csil-cpu8:~/Downloads$ ls
file.txt
sepidh@csil-cpu8:~/Downloads$ cd ..
sepidh@csil-cpu8:~$ mv Downloads_p Do
Documents/   Downloads/   Downloads_p/
sepidh@csil-cpu8:~$ mv Downloads_p Downloads
sepidh@csil-cpu8:~$ ls
 Android   Documents   file2.txt   Pictures   sfuhome     Videos
 Desktop   Downloads   Music       Public     Templates  'VirtualBox VMs'
sepidh@csil-cpu8:~$ cd Downloads/
sepidh@csil-cpu8:~/Downloads$ ls
Downloads_p  file.txt
sepidh@csil-cpu8:~/Downloads$
```

Basic commands in Linux

- find: use to find files and directories.

```
sepidh@csil-cpu8:~$ ls
 Android     Downloads   Pictures    sfuhome    'VirtualBox VMs'
 Desktop     file2.txt   Public      Templates
 Documents   Music       sepid.txt   Videos
sepidh@csil-cpu8:~$ find sepid.txt
sepid.txt
sepidh@csil-cpu8:~$ find *.txt
file2.txt
sepid.txt
sepidh@csil-cpu8:~$
```

Basic commands in Linux

- grep: use it to search in the notepad file. Lines that contain the searched word will be displayed fully.

```
Hi my name is Sepid.
I am SFU students.
I study computing science.
```

```
sepidh@csil-cpu8:~$ grep stud sepid.txt
I am SFU students.
I study computing science.
sepidh@csil-cpu8:~$
```

Basic commands in Linux

- man: use it to view the reference manuals of a command or utility run in the terminal. The output of the command displays the available man page headings for the specified command.

The list of possible headings includes:

- **Name: The name of the command.**
- **Synopsis: The command's syntax.**
- **Configuration: Configuration details for a device.**
- **Description: A description of the command.**
- **Examples: Several examples demonstrating the use of the command.**
- **Defaults: The default functions of the command and how they can be overridden.**

In the terminal window, type man followed by the Linux command name which man page you want to see.

The output is lengthy. Use the mouse scroll wheel, the up and down arrow keys, or the PgDn and PgUp keys to navigate through it.

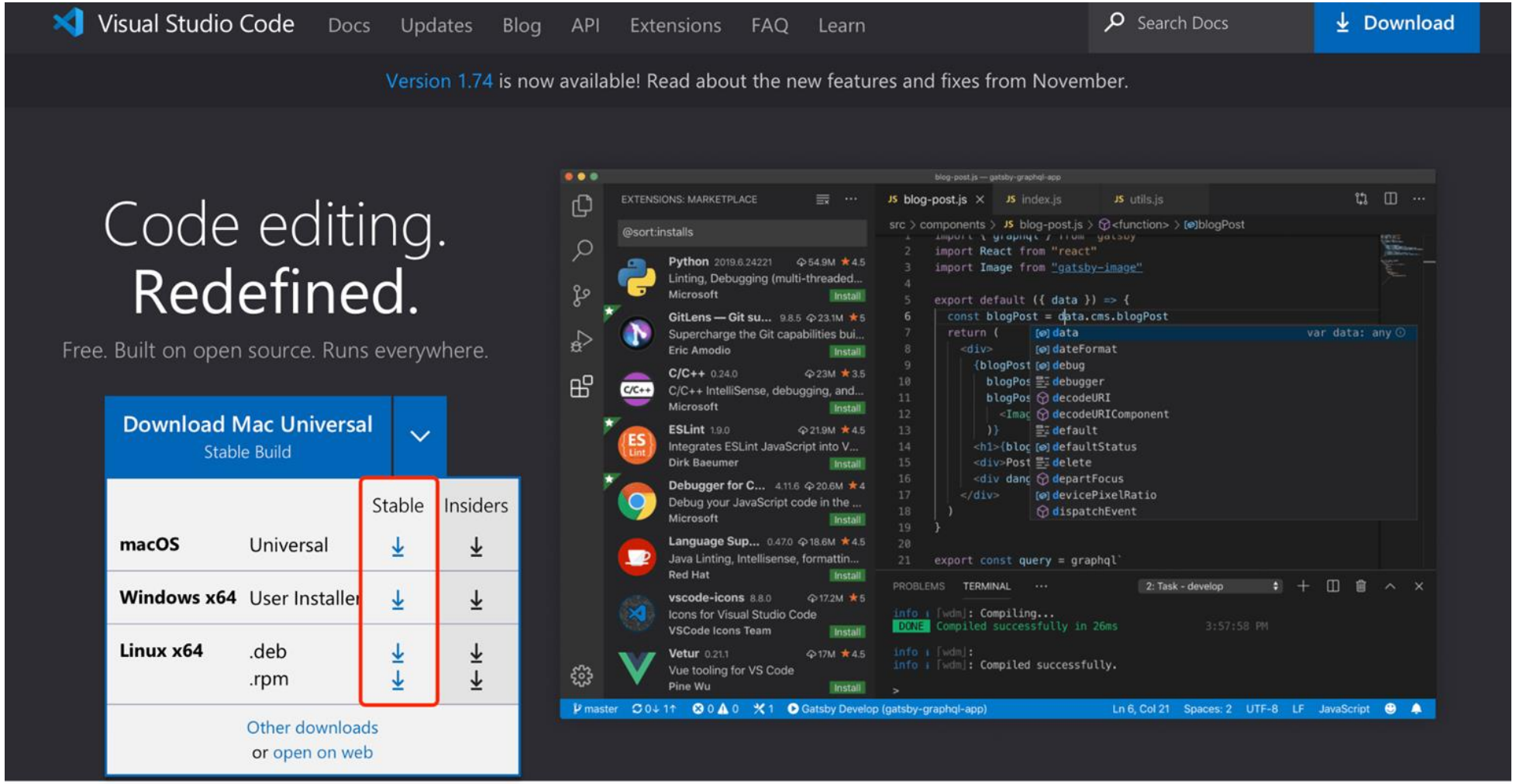
To exit, press Q.

output for man rm:

```
RM(1)                    General Commands Manual                   RM(1)

NAME
     rm, unlink — remove directory entries

SYNOPSIS
     rm [-f | -i] [-dIRrvWx] file ...
     unlink [--] file

DESCRIPTION
     The rm utility attempts to remove the non-directory type files specified on
     the command line.  If the permissions of the file do not permit writing,
     and the standard input device is a terminal, the user is prompted (on the
     standard error output) for confirmation.

     The options are as follows:

     -d      Attempt to remove directories as well as other types of files.

     -f      Attempt to remove the files without prompting for confirmation,
             regardless of the file's permissions.  If the file does not exist,
             do not display a diagnostic message or modify the exit status to
             reflect an error.  The -f option overrides any previous -i options.
:
```

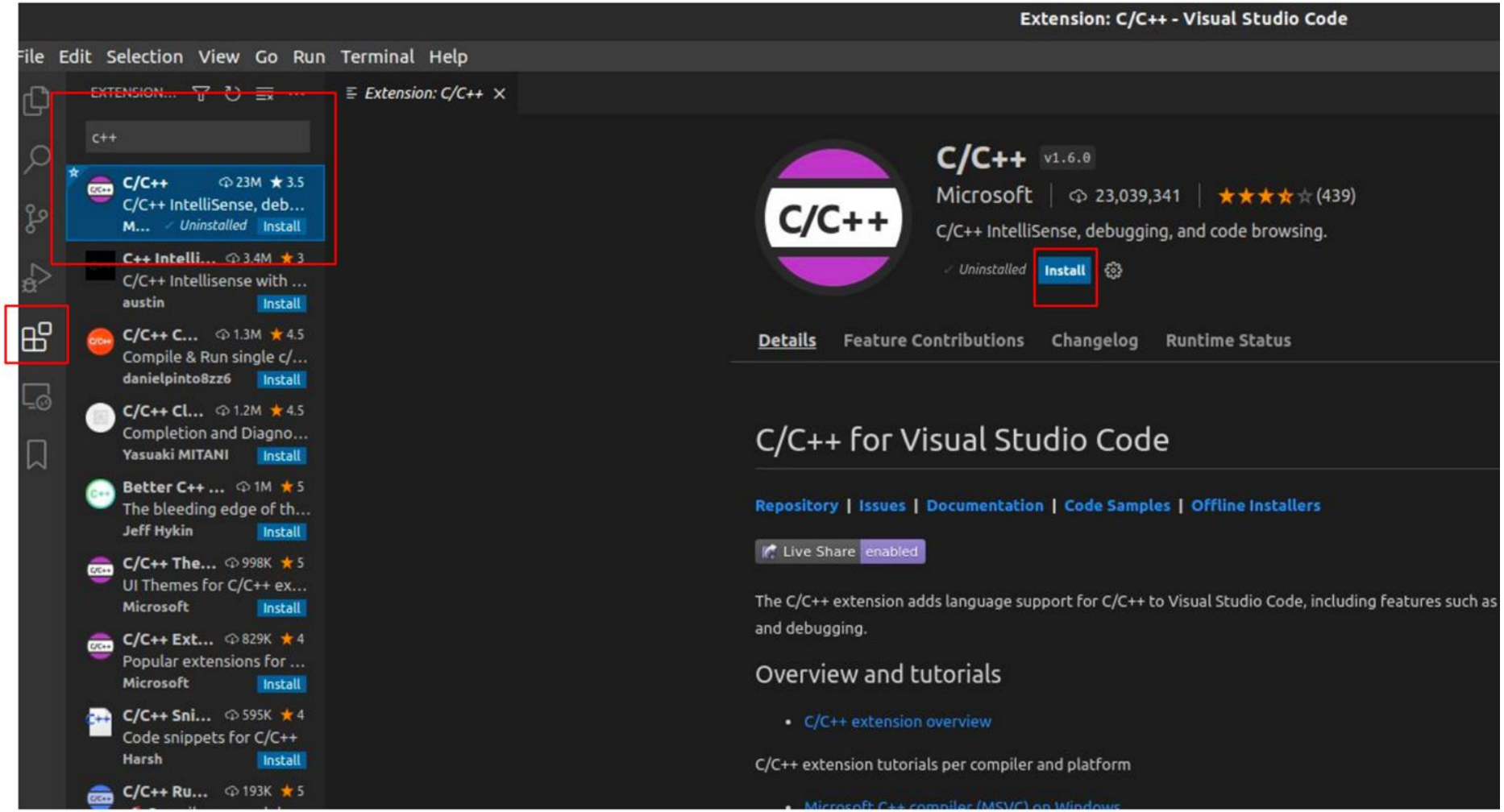
VS Code configuration

- download and start the app: https://code.visualstudio.com



VS Code configuration

- install c/c++ extension



Hello World (VS Code)

- Create a Hello World source code called hello.c

go to File, create new file. paste “hello world” program below in the file and save is as hello.c on your machine.

hello world:

```c
#include <stdio.h>
int main() {
      // printf() displays the string inside quotation
      printf("Hello, World!\n");
      return 0;
}
```

Compile and run your code

If you are in CSIL, you have two main ways to compile and run your code:

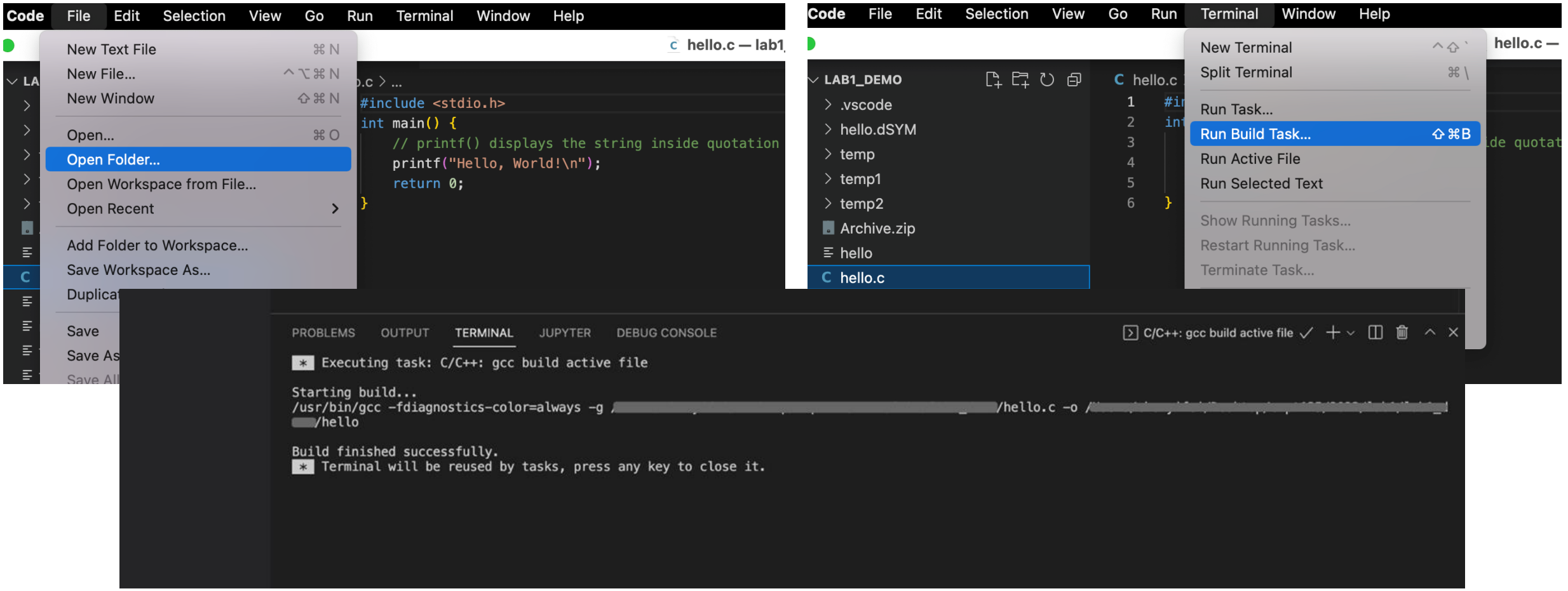
- **using VS Code and its own C/C++ compiler**
- **using the terminal app and CSIL's C/C++ compiler**

If you are no in CSIL, you also have two main ways to compile and run your code, provided that you are either still in campus, or connect to SFU VPN when outside campus:

- **using VS Code and its SSH tool**
- **using the terminal app after creating an SSH connection**

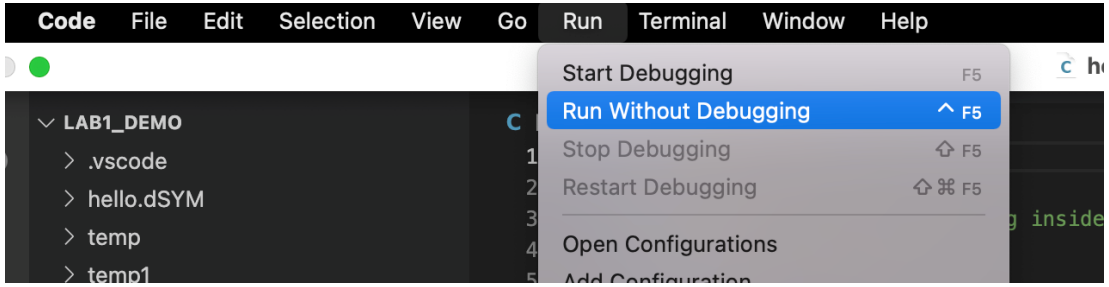
Compile and run your code **using VS Code and its own C/C++ compiler**:

- Open the folder where your file is in
- From terminal choose “Run Build Task”, then choose “C/C++: gcc build active file”. Now your code is ready to run and debug using



Compile and run your code **using VS Code and its own C/C++ compiler**:

- now your code is ready to run and debug using “Run” in VS code.

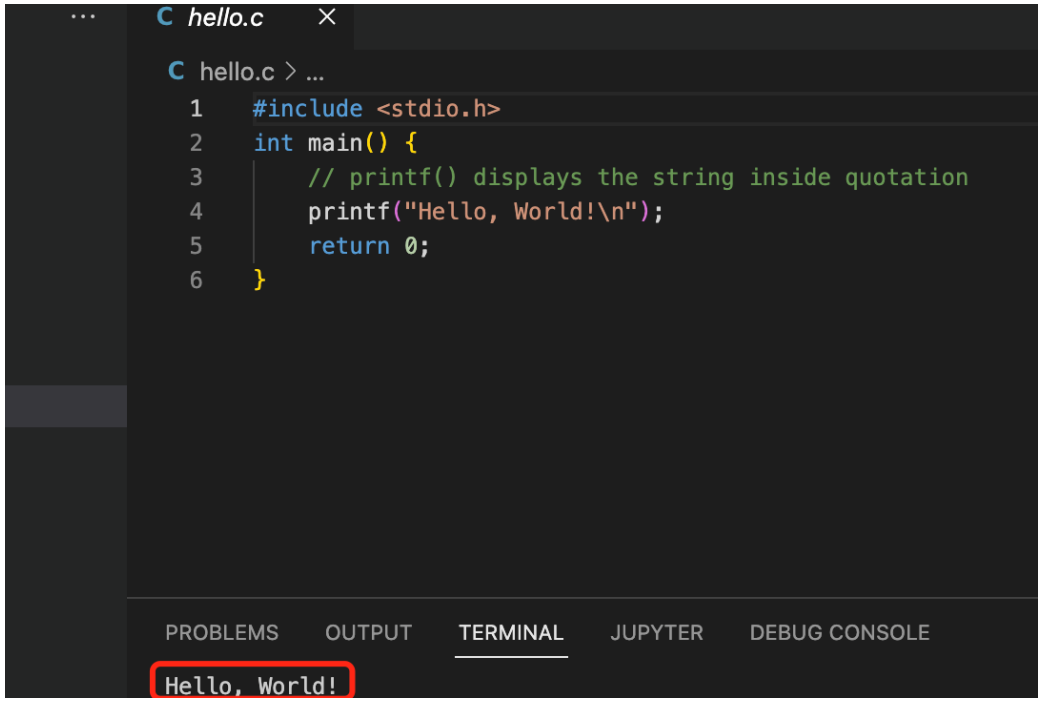


In “Run” menu at top choose “Run Without Debugging”.

Then choose C++(GDB, LLDB), and then choose “gcc build and debug active file”.

At terminal down you should be able to see output.

Change “Hello, World!\n” to any sentence you like and see how printed output changes.



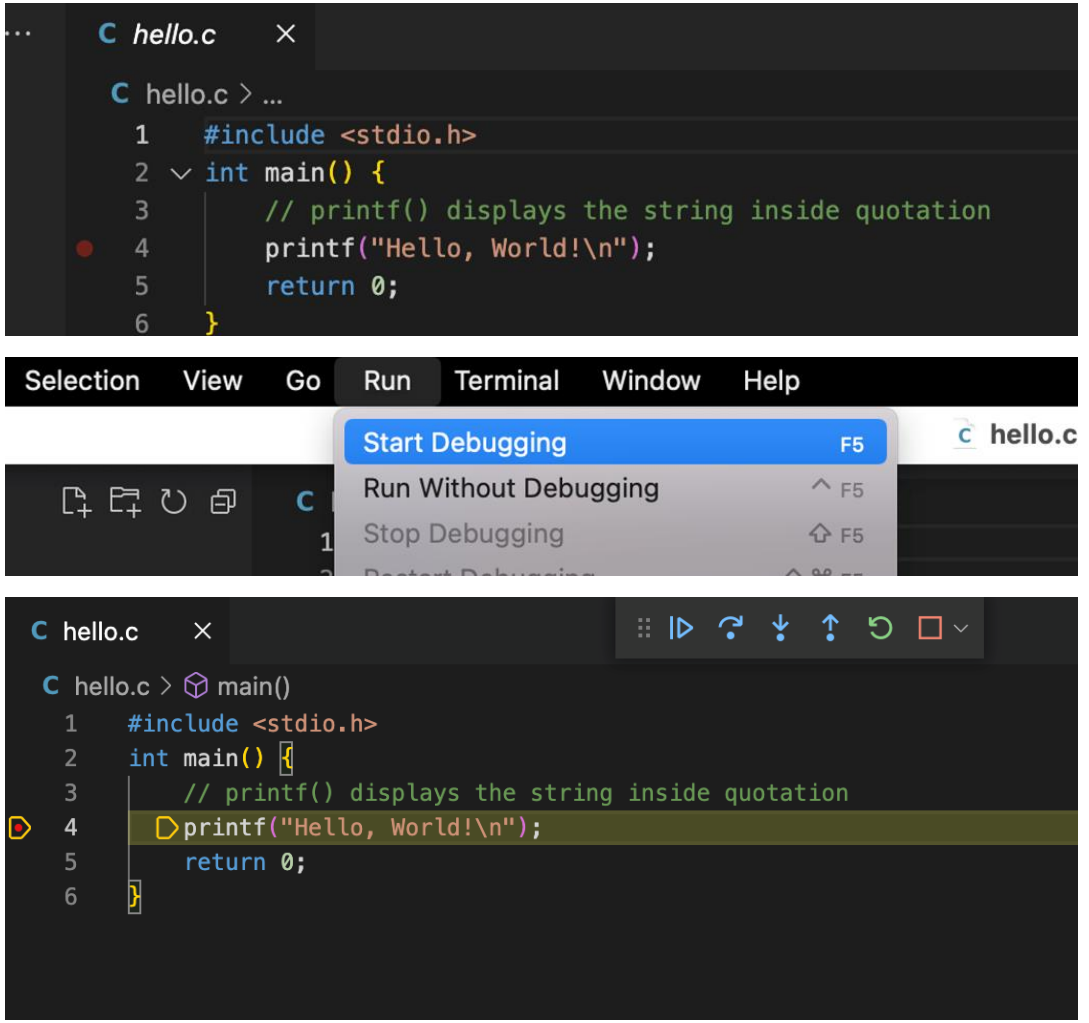
Compile and run your code **using VS Code and its own C/C++ compiler**:

- now your code is ready to run and debug using “Run” in VS code.

To debug the code first click on the lines that you want your code stop on. you should see small red circles there after clicking.

Then from Run menu choose Start Debugging.
It should stop in the selected lines.

Then by using options at top you can navigate through your code and move inside your code



Compile and run your code **using the terminal app and CSIL's C/C++ compiler**

1. open a terminal and go the the directory that your file is in.

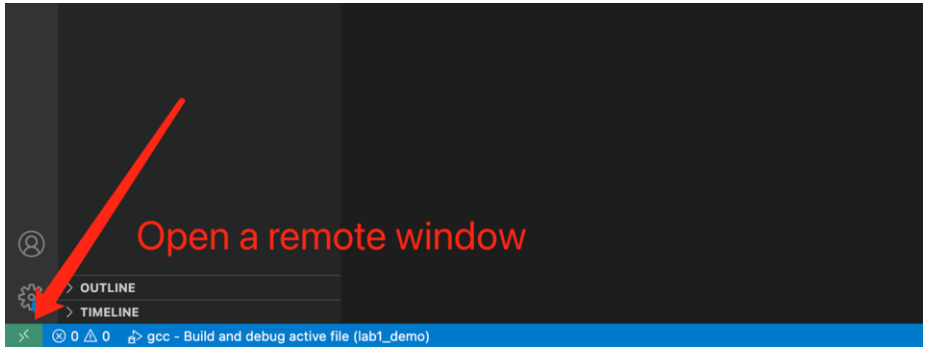
2. use cp to create another Hello World source code called helloCL.c:
cp hello.c helloCL.c

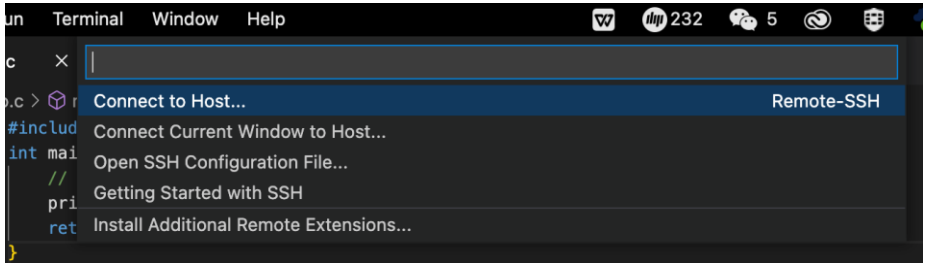
3. to create an executable file for file helloCL.c, and name the executable helloWorld, write this command in the terminal:
gcc helloCL.c -o helloWorld

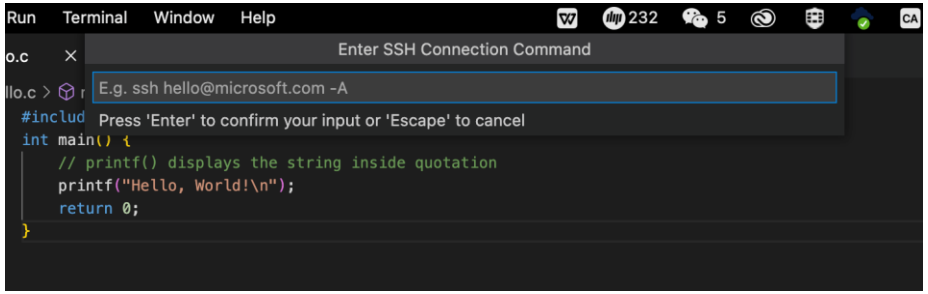
- **-o sets the name of the output file that GCC produces. You're using it when linking object files to make a complete program, and the default output filename for that is a.out. If you don't want your program to be called a.out, you use -o to specify a different name.**

4. run the executable with this command in the terminal:
./helloWorld

Compile and run your code **using VS Code and its SSH tool**







Using VS code (connect to SFU VPN from your local computer)

1. Open a remote window in VS code

2. Choose “Connect to Host”, and “add new SSH host”

3. connect to the host with:
ssh -p24 *your_sfu_computing_id@csil_host*.csil.sfu.ca

where,

your_sfu_computing_id: your SFU Computing ID

csil_host: one of the CSIL hosts listed in:
https://www.sfu.ca/computing/about/support/covid-19-response--working-remotely/csil-linux-remote-access.html#csil-linux-systems

4. Once connect to the host, you can open/move/edit/copy/etc. your files the same way as on your own computer

Compile and run your code **using the terminal app after creating an SSH connection**

Using VS code (connect to SFU VPN from your local computer)

1. Open your terminal app

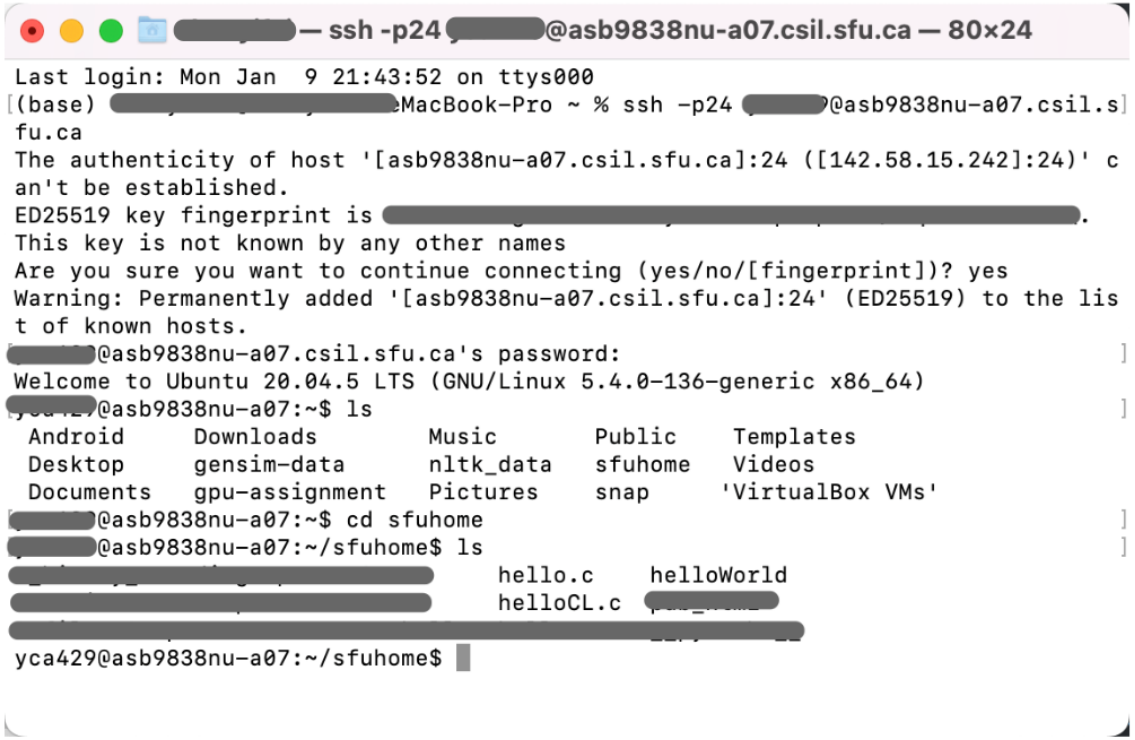
2. connect to the host with:

ssh -p24 *your_sfu_computing_id@csil_host*.csil.sfu.ca

where,

your_sfu_computing_id: your SFU Computing ID

csil_host: one of the CSIL hosts listed in: https://www.sfu.ca/computing/about/support/covid-19-response--working-remotely/csil-linux-remote-access.html#csil-linux-systems



The way we recommend for assignments…

Using CSIL's own gcc compiler in terminal, with VS Code connect to CSIL host: