

6CS005 High Performance Computing

Pre-Requisite Workshop 2 Part 1: Compile and Run C Program in Ubuntu

Jnaneshwar Bohara

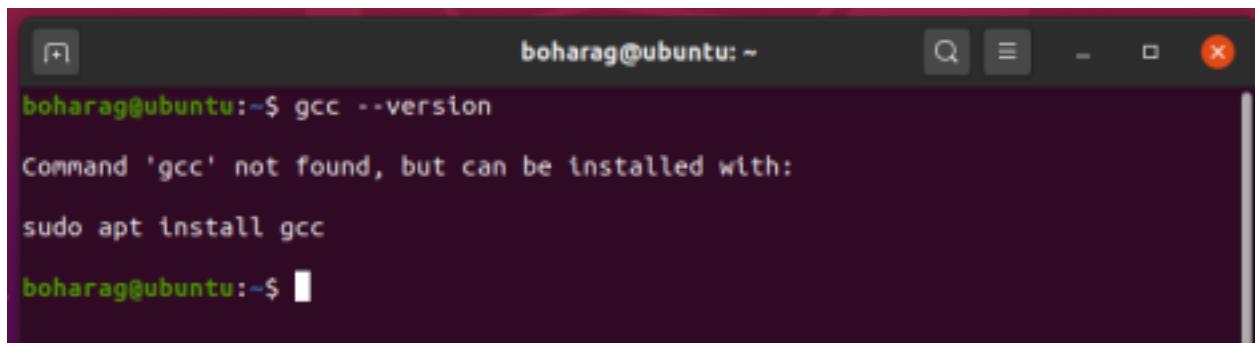
University of Wolverhampton

Herald College, Kathmandu

September 2024

Install GCC in Ubuntu

1. First check if you have gcc installed with command gcc --version

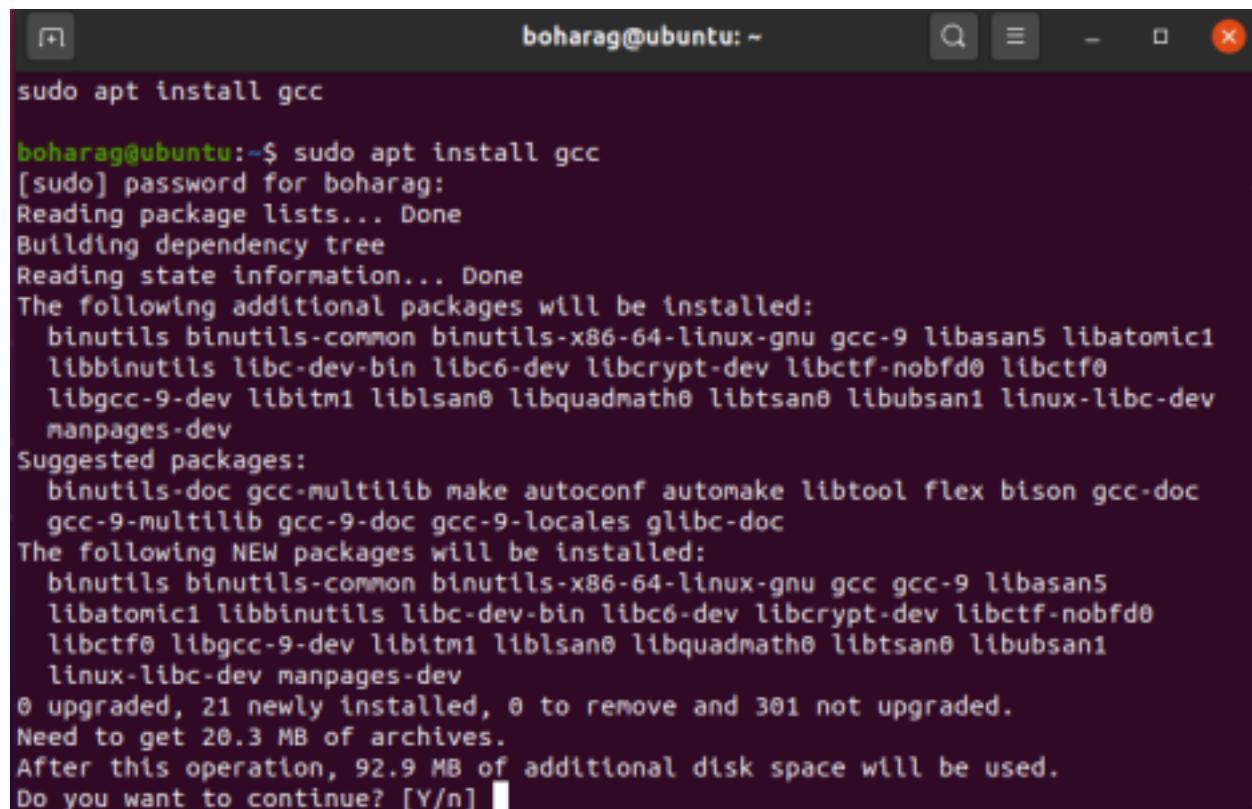


```
boharag@ubuntu:~$ gcc --version
Command 'gcc' not found, but can be installed with:
sudo apt install gcc
boharag@ubuntu:~$
```

A screenshot of a terminal window titled 'boharag@ubuntu: ~'. The window shows the command 'gcc --version' being run and its output. The output indicates that the command 'gcc' was not found, but it can be installed using the command 'sudo apt install gcc'. The terminal has a dark background with light-colored text.

2. If gcc is installed it will display the version else it will give you the command to install gcc sudo apt install gcc

3. Now execute the command ‘sudo apt install gcc’ and it will proceed as shown:



```
boharag@ubuntu:~$ sudo apt install gcc
[sudo] password for boharag:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu gcc-9 libasan5 libatomic1
  libbinutils libc-dev-bin libc6-dev libcrypt-dev libctf-nobfd0 libctf0
  libgcc-9-dev libitm1 liblsan0 libquadmath0 libtsan0 libubsan1 linux-libc-dev
  manpages-dev
Suggested packages:
  binutils-doc gcc-multilib make autoconf automake libtool flex bison gcc-doc
  gcc-9-multilib gcc-9-doc gcc-9-locales glibc-doc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu gcc gcc-9 libasan5
  libatomic1 libbinutils libc-dev-bin libc6-dev libcrypt-dev libctf-nobfd0
  libctf0 libgcc-9-dev libitm1 liblsan0 libquadmath0 libtsan0 libubsan1
  linux-libc-dev manpages-dev
0 upgraded, 21 newly installed, 0 to remove and 301 not upgraded.
Need to get 20.3 MB of archives.
After this operation, 92.9 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

4. Select ‘Y’ to continue, it will proceed as shown:

```
buntu1 [46.6 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu focal/main amd64 binutils-x86-64-linux
-gnu amd64 2.34-6ubuntu1 [1,614 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu focal/main amd64 binutils amd64 2.34-6
ubuntu1 [3,376 B]
Get:7 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libitm1 amd64 10-2020
0411-0ubuntu1 [26.3 kB]
Get:8 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libatomic1 amd64 10-2
0200411-0ubuntu1 [9,284 B]
Get:9 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libasan5 amd64 9.3.0-
10ubuntu2 [395 kB]
Get:10 http://us.archive.ubuntu.com/ubuntu focal/main amd64 liblsan0 amd64 10-20
200411-0ubuntu1 [144 kB]
Get:11 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libtsan0 amd64 10-20
200411-0ubuntu1 [319 kB]
Get:12 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libubsan1 amd64 10-2
0200411-0ubuntu1 [136 kB]
Get:13 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libquadmath0 amd64 1
0-20200411-0ubuntu1 [146 kB]
Get:14 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libgcc-9-dev amd64 9
.3.0-10ubuntu2 [2,359 kB]
Get:15 http://us.archive.ubuntu.com/ubuntu focal/main amd64 gcc-9 amd64 9.3.0-10
ubuntu2 [8,234 kB]
37% [15 gcc-9 13.6 kB/8,234 kB 0%] 377 kB/s 38s
```

5. If everything goes fine it will be completed as:

```
boharag@ubuntu:~
Setting up linux-libc-dev:amd64 (5.4.0-47.51) ...
Setting up libctf-nobfd0:amd64 (2.34-6ubuntu1) ...
Setting up libasan5:amd64 (9.3.0-10ubuntu2) ...
Setting up libquadmath0:amd64 (10-20200411-0ubuntu1) ...
Setting up libatomic1:amd64 (10-20200411-0ubuntu1) ...
Setting up libubsan1:amd64 (10-20200411-0ubuntu1) ...
Setting up libcrypt-dev:amd64 (1:4.4.10-10ubuntu4) ...
Setting up libbinutils:amd64 (2.34-6ubuntu1) ...
Setting up libc-dev-bin (2.31-0ubuntu9) ...
Setting up liblsan0:amd64 (10-20200411-0ubuntu1) ...
Setting up libitm1:amd64 (10-20200411-0ubuntu1) ...
Setting up libtsan0:amd64 (10-20200411-0ubuntu1) ...
Setting up libctf0:amd64 (2.34-6ubuntu1) ...
Setting up libgcc-9-dev:amd64 (9.3.0-10ubuntu2) ...
Setting up libc6-dev:amd64 (2.31-0ubuntu9) ...
Setting up binutils-x86-64-linux-gnu (2.34-6ubuntu1) ...
Setting up binutils (2.34-6ubuntu1) ...
Setting up gcc-9 (9.3.0-10ubuntu2) ...
Setting up gcc (4:9.3.0-1ubuntu2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9) ...
boharag@ubuntu:~$
```

6. Now you can check whether gcc is installed successfully with command `gcc –version`. It will give the details of version installed.

The screenshot shows a terminal window titled "boharag@ubuntu:~". The terminal displays the output of the command `gcc --version`. The output includes the following text:
Setting up libatomic1:amd64 (10-20200411-0ubuntu1) ...
Setting up libubsan1:amd64 (10-20200411-0ubuntu1) ...
Setting up libcrypt-dev:amd64 (1:4.4.10-10ubuntu4) ...
Setting up libbinutils:amd64 (2.34-6ubuntu1) ...
Setting up libc-dev-bin (2.31-0ubuntu9) ...
Setting up liblsan0:amd64 (10-20200411-0ubuntu1) ...
Setting up libitm1:amd64 (10-20200411-0ubuntu1) ...
Setting up libtsan0:amd64 (10-20200411-0ubuntu1) ...
Setting up libctf0:amd64 (2.34-6ubuntu1) ...
Setting up libgcc-9-dev:amd64 (9.3.0-10ubuntu2) ...
Setting up libc6-dev:amd64 (2.31-0ubuntu9) ...
Setting up binutils-x86-64-linux-gnu (2.34-6ubuntu1) ...
Setting up binutils (2.34-6ubuntu1) ...
Setting up gcc-9 (9.3.0-10ubuntu2) ...
Setting up gcc (4:9.3.0-1ubuntu2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9) ...
boharag@ubuntu:~\$ gcc --version
gcc (Ubuntu 9.3.0-10ubuntu2) 9.3.0
Copyright (C) 2019 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
boharag@ubuntu:~\$

7. Now you have successfully installed gcc and you are ready to compile your C programs.

Compile and Run C Program

1. Use a text editor to write the C source code.

For example, type the command below command to create the file hello.c
`gedit hello.c`

It will open empty file with name hello.c for you to enter the source code.

2. Now enter the C source code below:

```
#include <stdio.h>

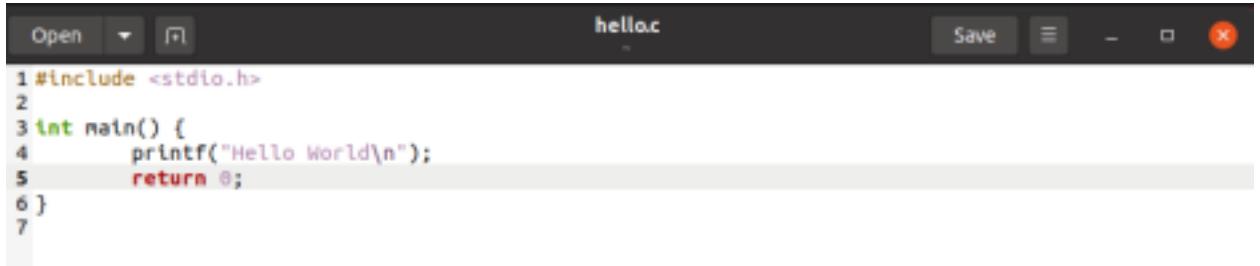
int main() {

    printf("Hello World\n");

    return 0;

}
```

3. Close the editor window



```
hello.c
1 #include <stdio.h>
2
3 int main() {
4     printf("Hello World\n");
5     return 0;
6 }
7
```

4. Compile the program with below command:

```
gcc hello.c -o hello
```

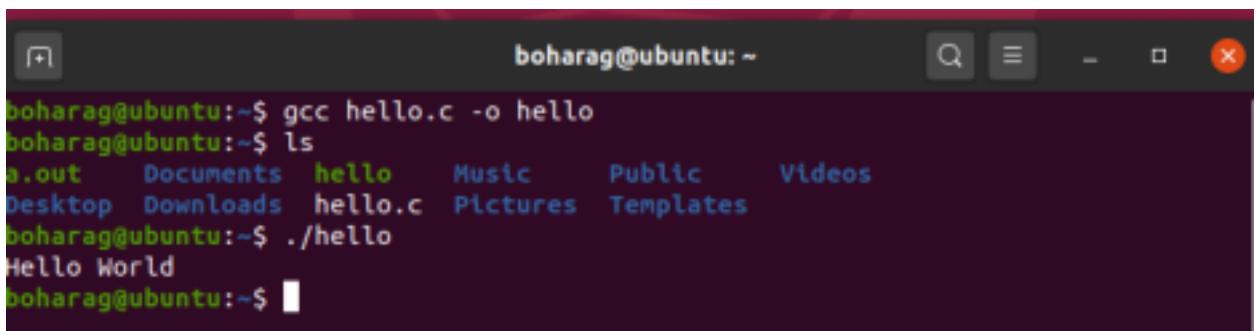
This command will invoke the GNU C compiler to compile the file hello.c and output (-o) the result to an executable called hello.

5. Now execute the program with below command:

```
./hello
```

This should result in the output

```
Hello World
```



```
boharag@ubuntu:~$ gcc hello.c -o hello
boharag@ubuntu:~$ ls
a.out    Documents  hello    Music      Public      Videos
Desktop  Downloads  hello.c  Pictures   Templates
boharag@ubuntu:~$ ./hello
Hello World
boharag@ubuntu:~$
```

5 | Compile and Run C Program in Ubuntu – Gyaneshwar Bohara