

EXNO: 6

DATE:

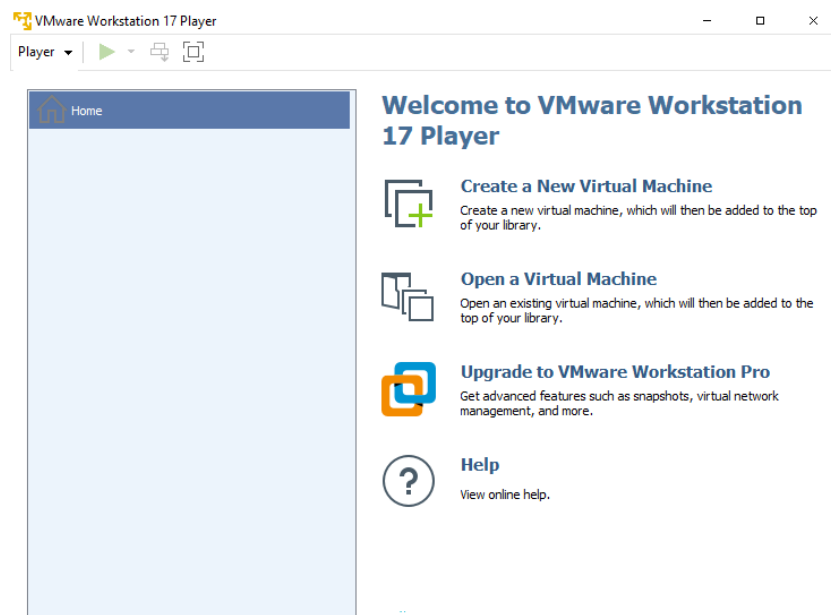
Install C compiler in the Virtual Machine and execute a Sample Program

AIM:

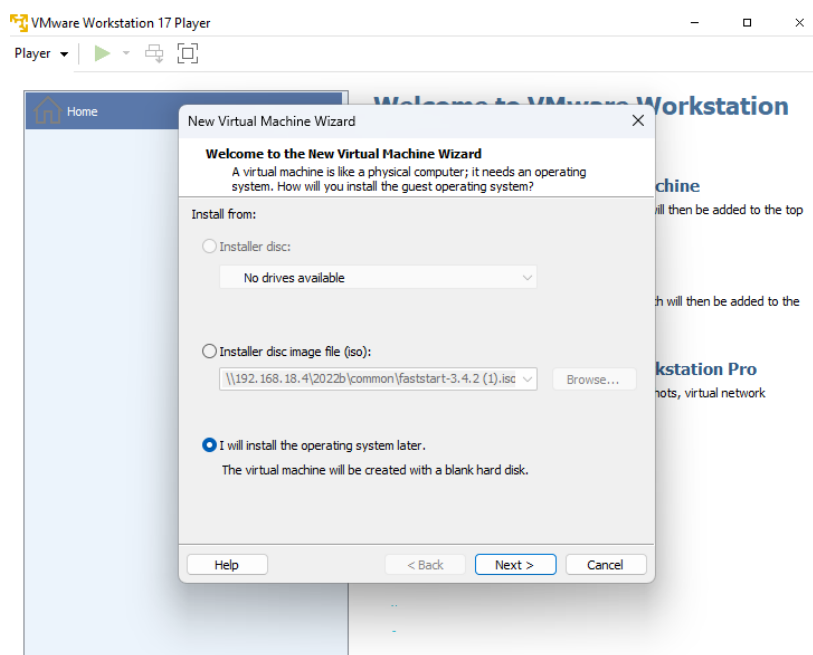
To implement the installation of C compiler in the virtual machine and execution of a sample program.

PROCEDURE:

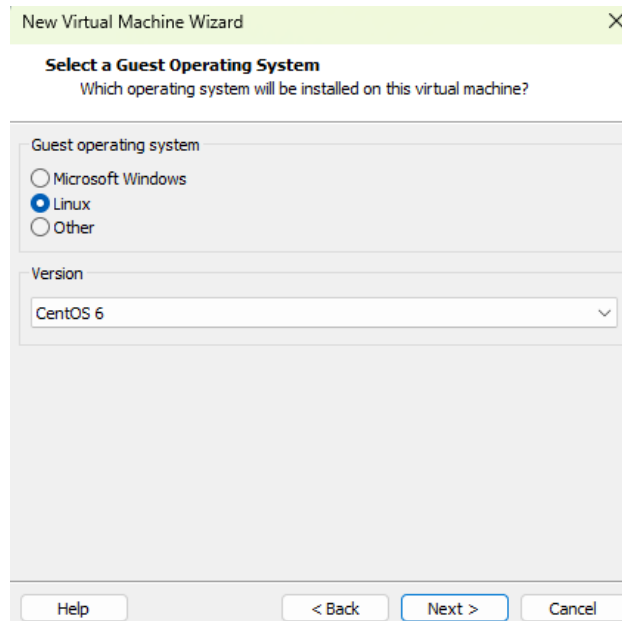
1. Click on VMWare workstation in your desktop and click on Create a New Virtual Machine.



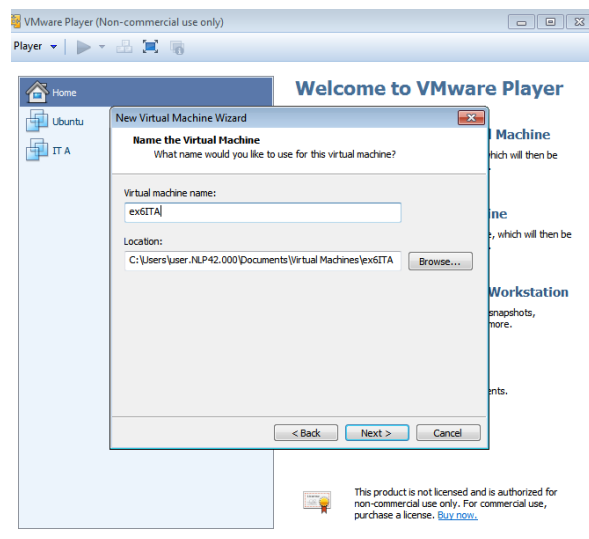
2. Choose the 'I will install the operating system later' option and click Next.



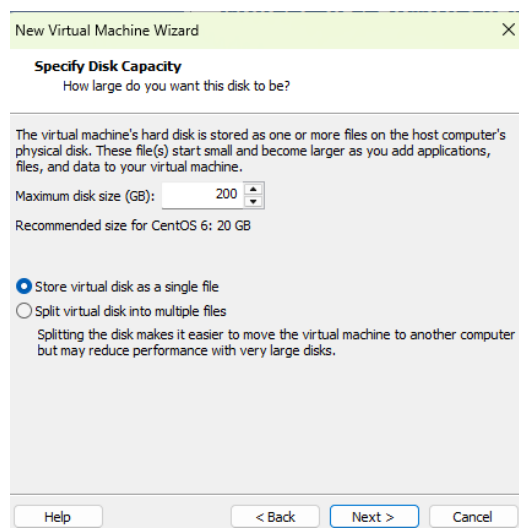
3. Choose Linux as the Guest Operating system, select CentOS 6 as version and click Next



4. Name the Virtual Machine as ex6ITA, click Next.

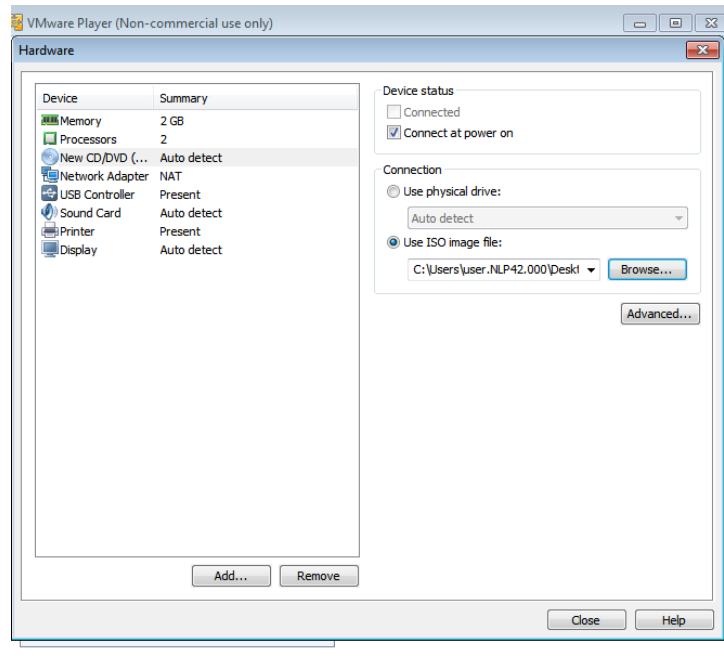


5. Enter the maximum disk size as 200GB and choose the option .Store visual disk as a single file.

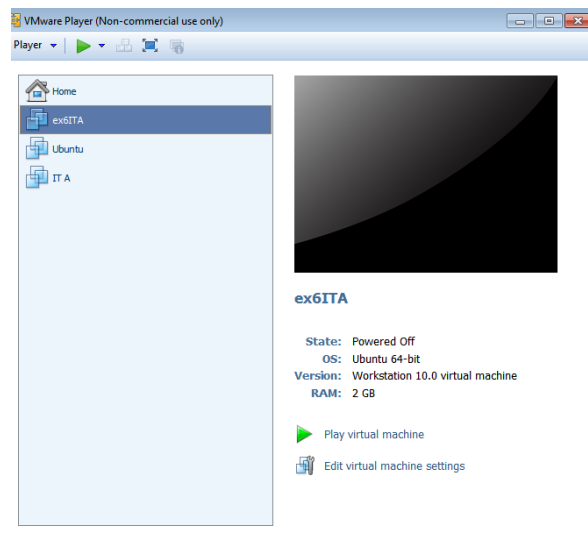


6. Click on Customize Hardware and configure the Virtual Machine as below:

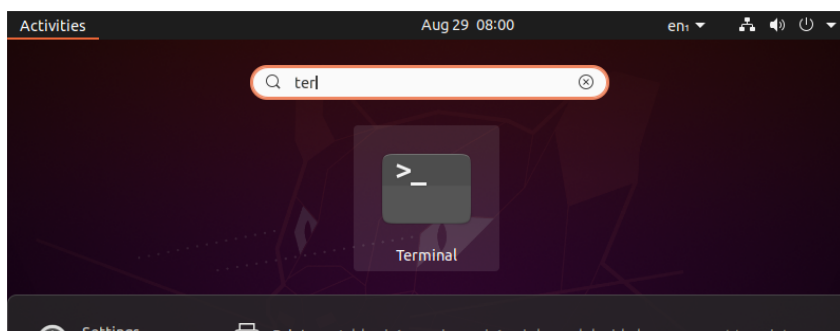
- i) Choose the memory as 2GB(2048 MB)
- ii) Enter the number of processors as 2.
- iii) Choose ISO image file, click Browse or search, AMD64.iso, click close and then finish.



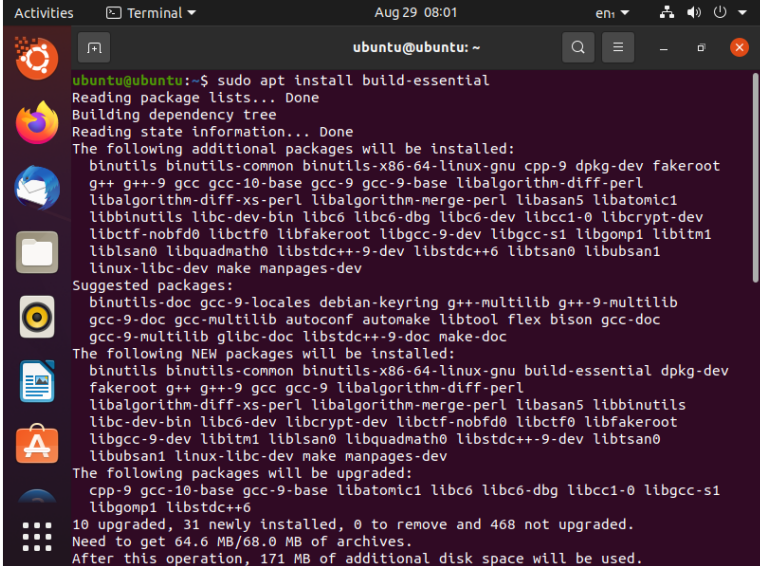
7. Click Play Virtual Machine.



8. Open the terminal in Ubuntu OS.

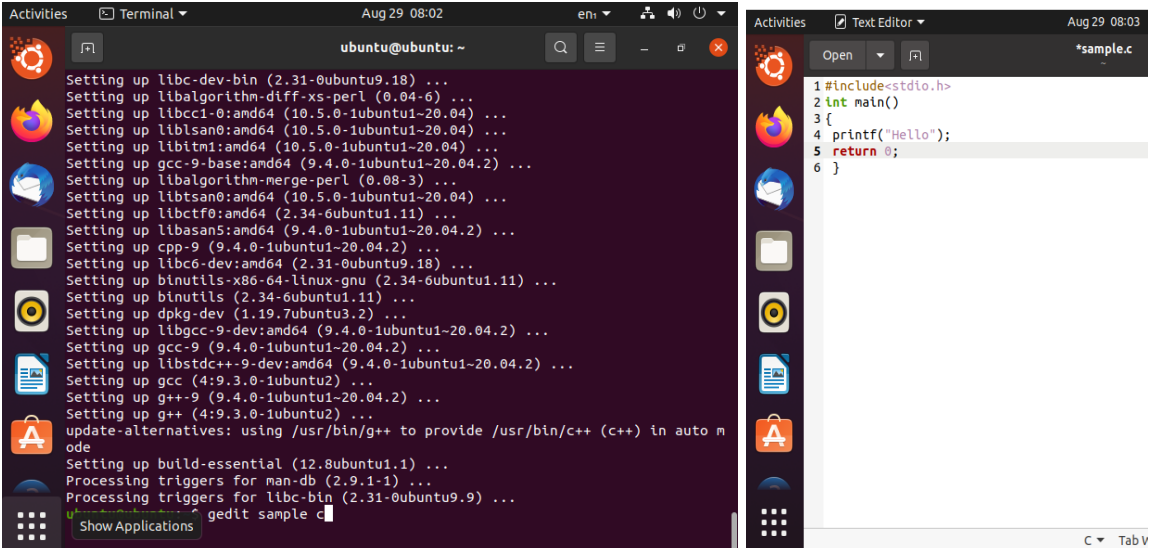


8. Installed C compiler using the command “sudo apt install build-essential” with password “123456”.



```
ubuntu@ubuntu:~$ sudo apt install build-essential
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 cpp-9 dpkg-dev fakeroot
  g++ g++-9 gcc gcc-10-base gcc-9 gcc-9-base libalgorithm-diff-perl
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5 libatomic1
  libbinutils libc-dev-bin libc6 libc6-dbg libc6-dev libcc1-0 libcrypt-dev
  libctf-nobfd0 libctf0 libfakeroot libgcc-9-dev libgcc-s1 libgomp1 libitm1
  liblsan0 libquadmath0 libstdc++-9-dev libstdc++6 libtsan0 libubsan1
  linux-libc-dev make manpages-dev
Suggested packages:
  binutils-doc gcc-9-locales debian-keyring g++-multilib g++-9-multilib
  gcc-9-doc gcc-multilib autoconf automake libtool flex bison gcc-doc
  gcc-9-multilib glibc-doc libstdc++-9-doc make-doc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev
  fakeroot g++ g++-9 gcc gcc-9-base libalgorithm-diff-perl
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5 libbinutils
  libc-dev-bin libc6-dev libcrypt-dev libctf-nobfd0 libctf0 libfakeroot
  libgcc-9-dev libitm1 liblsan0 libquadmath0 libstdc++-9-dev libtsan0
  libubsan1 linux-libc-dev make manpages-dev
The following packages will be upgraded:
  cpp-9 gcc-10-base gcc-9-base libatomic1 libc6 libc6-dbg libcc1-0 libgcc-s1
  libgomp1 libstdc++6
10 upgraded, 31 newly installed, 0 to remove and 468 not upgraded.
Need to get 64.6 MB/68.0 MB of archives.
After this operation, 171 MB of additional disk space will be used.
```

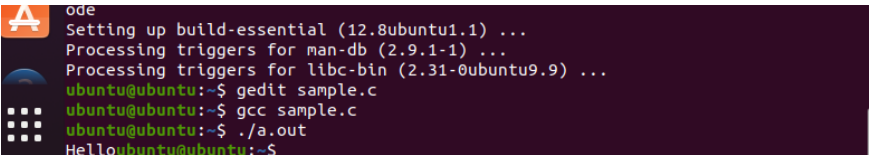
9. Create a sample C program using the command “gedit”.



```
Setting up libc-dev-bin (2.31-0ubuntu9.18) ...
Setting up libalgorithm-diff-xs-perl (0.04-6) ...
Setting up libcc1-0:amd64 (10.5.0-1ubuntu1-20.04) ...
Setting up liblsan0:amd64 (10.5.0-1ubuntu1-20.04) ...
Setting up libitm1:amd64 (10.5.0-1ubuntu1-20.04) ...
Setting up gcc-9-base:amd64 (9.4.0-1ubuntu1-20.04.2) ...
Setting up libalgorithm-merge-perl (0.08-3) ...
Setting up libtsan0:amd64 (10.5.0-1ubuntu1-20.04) ...
Setting up libasan5:amd64 (9.4.0-1ubuntu1-20.04.2) ...
Setting up cpp-9 (9.4.0-1ubuntu1-20.04.2) ...
Setting up libc6-dev:amd64 (2.31-0ubuntu9.18) ...
Setting up binutils-x86-64-linux-gnu (2.34-6ubuntu1.11) ...
Setting up binutils (2.34-6ubuntu1.11) ...
Setting up dpkg-dev (1.19.7ubuntu3.2) ...
Setting up libgcc-9-dev:amd64 (9.4.0-1ubuntu1-20.04.2) ...
Setting up gcc-9 (9.4.0-1ubuntu1-20.04.2) ...
Setting up libstdc++-9-dev:amd64 (9.4.0-1ubuntu1-20.04.2) ...
Setting up gcc (4:9.3.0-1ubuntu2) ...
Setting up g++-9 (9.4.0-1ubuntu1-20.04.2) ...
Setting up g++ (4:9.3.0-1ubuntu2) ...
update-alternatives: using /usr/bin/g++ to provide /usr/bin/c++ (c++) in auto m
ode
Setting up build-essential (12.8ubuntu1.1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
ubuntu@ubuntu:~$ gedit sample.c
```

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

10. Compile the C program using the command “gcc file_name”.
11. Run the compiled code using the command “./a.out”.



```
ubuntu@ubuntu:~$ gcc sample.c
ubuntu@ubuntu:~$ ./a.out
Hello
```

RESULT: Thus, Google App Engine has been installed and Hello world and other simple web applications have been created using Python/Java.