Task 1

1) Bash Script:

```
cd ./programs

for
file in *.py
do
mv "$file" "${file%.py}.c" done
```

2)

```
cd ./programs
echo "The programs available are: "

ls

read -p " Enter the program which requires compile and run " file

gcc "$file" -o "$file.out"

gnome-terminal -- bash -c " "./$file.out"; bash"

echo "done"
```

3) ROS Installation

```
echo "ROS installation has started"

sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release
-sc) main" > /etc/apt/sources.list.d/ros-latest.list'

sudo apt install curl

curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc |
sudo apt-key add -
sudo apt update

sudo apt install ros-melodic-desktop-full

echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc
source ~/.bashrc


sudo apt install python-rosdep python-rosinstall
python-rosinstall-generator python-wstool build-essential
sudo apt install python-rosdep

sudo rosdep init
rosdep update

echo "done"
```

4) Enclosing characters in single quotes (') preserves the literal value of each character within the quotes. A single quote may not occur between single quotes, even when preceded by a backslash.
Example: if a=abcd , this implies the expression '$a' will result in $a as, it has no specific meaning.
Enclosing characters in double quotes (") preserves the literal value of all characters within the quotes, with the exception of $, `, \, and, when history expansion is enabled, !. The characters $ and ` retain their special meaning within double quotes.
Example: if a=abcd , this implies the expression '$a' will result in abcd while considering it as a variable and expanding its value.

5) Export is a built-in bash function that can be used to declare a variable from the current shell, such that it can be used in any process within the shell. Declaring a variable without shell will make it so that the variable is available only to the shell and not the processes. This being a built-in function, gets directly interpreted by bash and thus can be used in external programs only within bash -c "export sample=1; echo $sample".
Now, in the given case when 'export rovername=vajra' is executed, even when a new shell session is started with $bash the variable rovername is recognised to have the value vajra, and echo $rovername will give vajra as output. Whereas when 'rovername=vajra' is executed, after changing the shell session the variable will not

be recognised, and if we try echo $rovername there'll be a blank output.

6)

```bash
#!/bin/bash

git clone https://github.com/ros/ros_tutorials.git

cd ./ros_tutorials

grep -R "3.14159"
```