

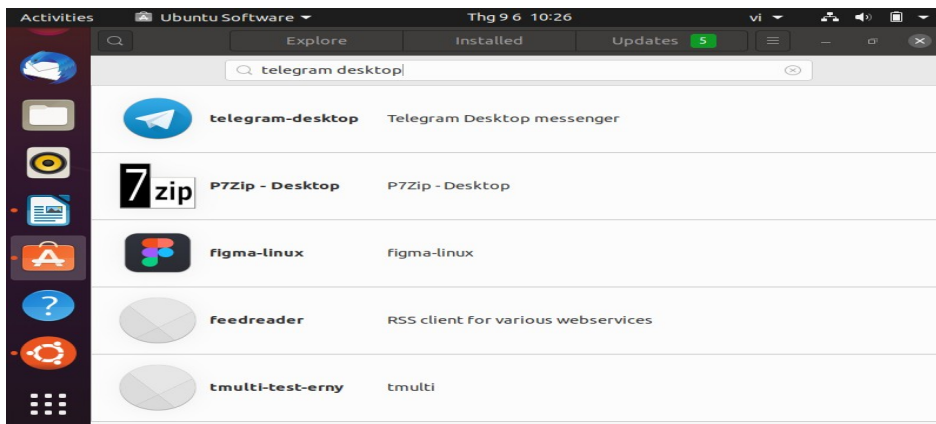
Họ tên: Trần Thị Huỳnh Như
MSSV: B1809385

CHỦ ĐỀ: MÔ HÌNH MẠNG TCP/IP

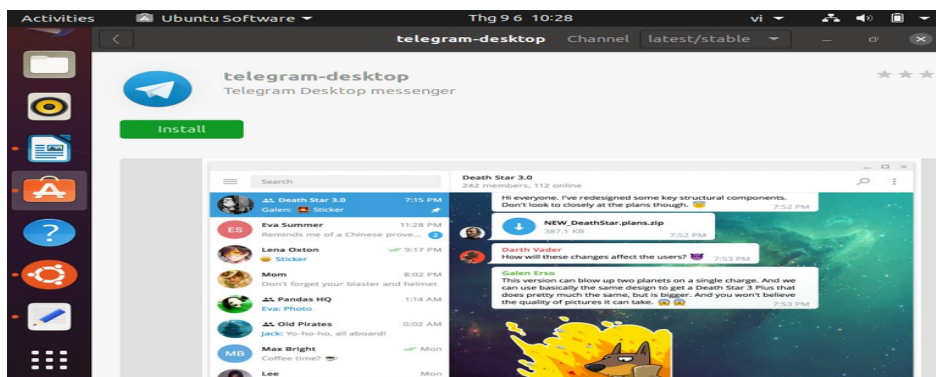
Chủ điểm: Cài đặt Ubuntu Desktop

Câu 5: Cài đặt Telegram Desktop từ kho phần mềm Ubuntu Software Center:

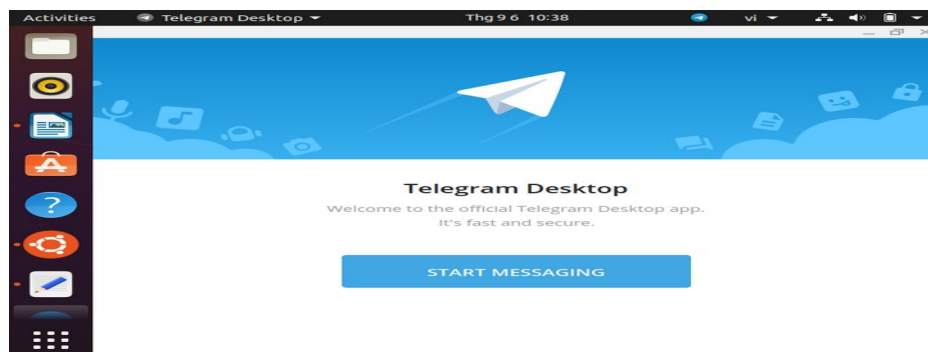
- Bước 1: Vào ubuntu software → gõ: “telegram desktop”. Chọn biểu tượng đầu tiên



- Bước 2: Nhấn install để cài đặt

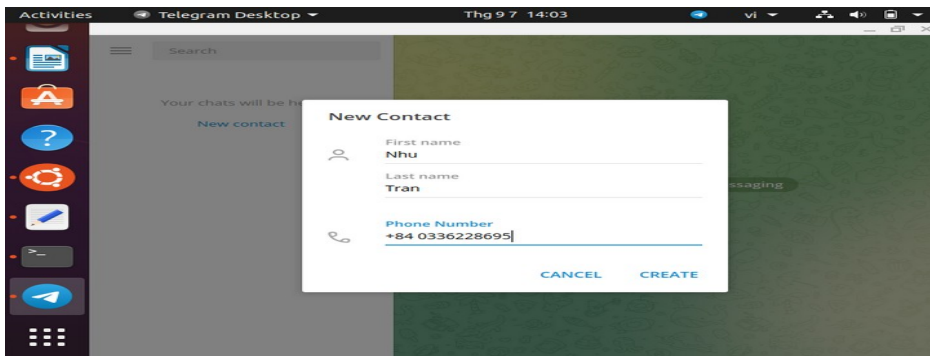


- Bước 3: Màn hình sau khi cài đặt thành công

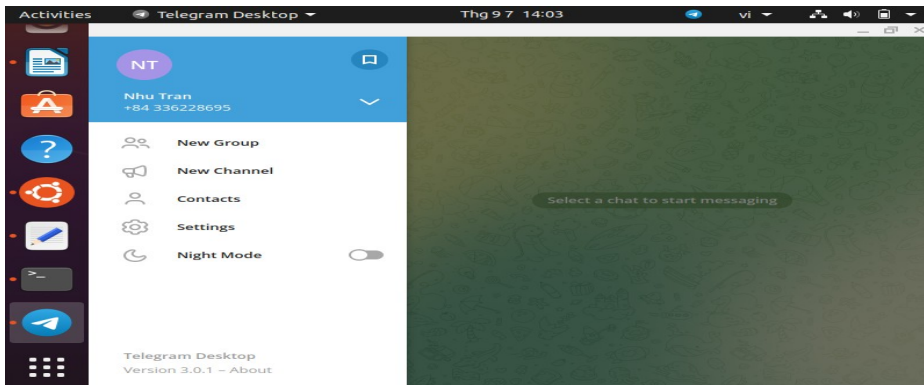


- Bước 4: Tiến hành cấu hình

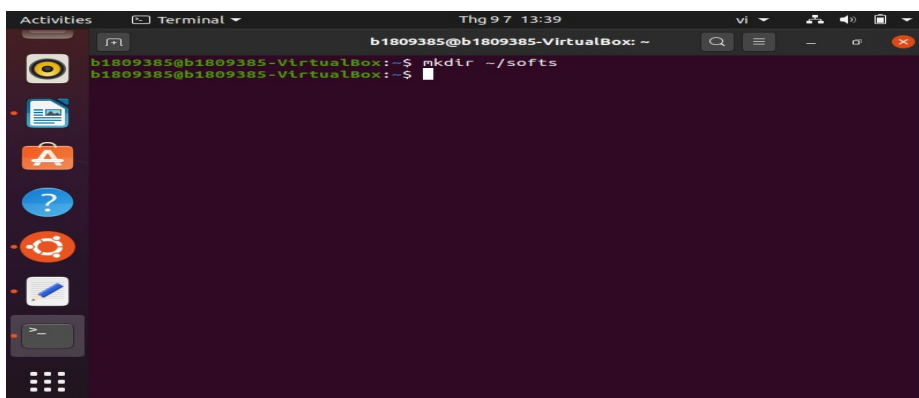
+ Nhập Tên và Họ và số điện thoại đăng kí



+ Màn hình khi cấu hình thành công

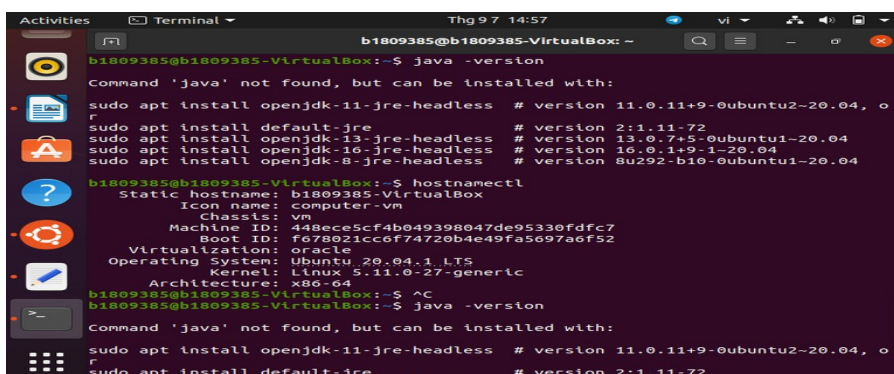


Câu 6: Trong thư mục cá nhân, tạo thư mục ~/softs:

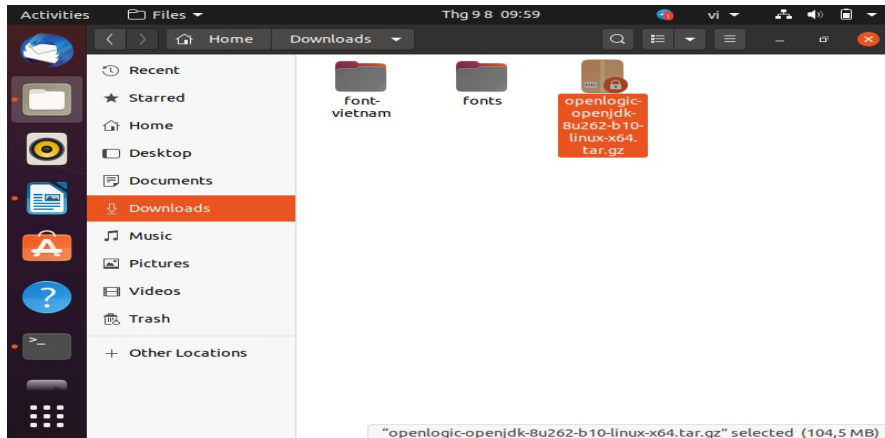


Câu 7: Download bộ phát triển ứng dụng bằng ngôn ngữ Java JDK, giải nén vào thư mục ~/softs:

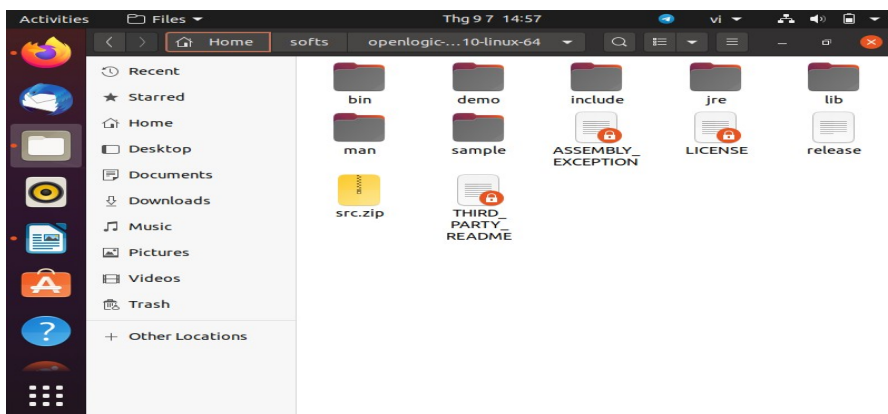
- Khi muốn kiểm tra phiên bản java đã được cài chưa gõ: `java -version`. Lúc này trên ảnh bên dưới java chưa được cài đặt
- Muốn chọn bộ JDK phù hợp cần kiểm tra phiên bản linux hiện tại, ta gõ lệnh sau : `hostnamectl`



- Sau khi biết được phiên bản, ta download phiên bản JDK phù hợp:



- Hình ảnh thư mục sau khi tải jdk → giải nén → đưa vào ~/softs:



Câu 8: Đặt biến môi trường đến các thư mục phù hợp của JDK:

- Tạo biến JAVA_HOME chứa đường dẫn trong ~/softs trở tới thư mục jdk vừa download
- Khai báo biến môi trường JAVA_HOME trở tới thư mục **bin** trong PATH
- Kiểm tra lại phiên bản java đã có chưa bằng lệnh: *java -version*
- Lúc này jdk 1.8 đã được cài đặt

```

b1809385@b1809385-VirtualBox: ~
Machine ID: 448ece5cf4b049398047de95330fdcf7
Boot ID: f678021cc6f74720b4e49fa5697a6f52
Virtualization: oracle
Operating System: Ubuntu 20.04.1 LTS
Kernel: Linux 5.11.0-27-generic
Architecture: x86_64
b1809385@b1809385-VirtualBox:~$ ^C
b1809385@b1809385-VirtualBox:~$ java -version
Command 'java' not found, but can be installed with:
sudo apt install openjdk-11-jre-headless # version 11.0.11+9-0ubuntu2~20.04, o
r
sudo apt install default-jre # version 2:1.11-72
sudo apt install openjdk-13-jre-headless # version 13.0.7+5-0ubuntu1~20.04
sudo apt install openjdk-16-jre-headless # version 16.0.1+9-1~20.04
sudo apt install openjdk-8-jre-headless # version 8u292-b10-0ubuntu1~20.04
b1809385@b1809385-VirtualBox:~$ export JAVA_HOME="/home/b1809385/softs/openl
ogic-openjdk-8u262-b10-linux-64"
$: command not found
b1809385@b1809385-VirtualBox:~$ export JAVA_HOME="/home/b1809385/softs/openlogi
c-openjdk-8u262-b10-linux-64"
b1809385@b1809385-VirtualBox:~$ export PATH=$JAVA_HOME/bin:$PATH
b1809385@b1809385-VirtualBox:~$ java -version
openjdk version "1.8.0-262"
OpenJDK Runtime Environment (build 1.8.0-262-b10)
OpenJDK 64-Bit Server VM (build 25.71-b10, mixed mode)
b1809385@b1809385-VirtualBox:~$

```

Câu 9: Kiểm tra giá trị biến môi trường:

Thực hiện thông qua lệnh `echo $JAVA_HOME`

```
Activities Terminal Thg 9 7 14:59
b1809385@b1809385-VirtualBox: ~
Virtualization: oracle
Operating System: Ubuntu 20.04.1 LTS
Kernel: Linux 5.11.0-27-generic
Architecture: x86_64
b1809385@b1809385-VirtualBox:~$ ^C
b1809385@b1809385-VirtualBox:~$ java -version
Command 'java' not found, but can be installed with:
sudo apt install openjdk-11-jre-headless # version 11.0.11+9-0ubuntu2~20.04, o
sudo apt install default-jre # version 2:1.11-72
sudo apt install openjdk-13-jre-headless # version 13.0.7+5-0ubuntu1~20.04
sudo apt install openjdk-16-jre-headless # version 16.0.1+9-1~20.04
sudo apt install openjdk-8-jre-headless # version 8u292-b10-0ubuntu1~20.04
b1809385@b1809385-VirtualBox:~$ $ export JAVA_HOME="/home/b1809385/softs/openl
ogic-openjdk-8u262-b10-linux-64"
$: command not found
b1809385@b1809385-VirtualBox:~$ export JAVA_HOME="/home/b1809385/softs/openlogi
c-openjdk-8u262-b10-linux-64"
b1809385@b1809385-VirtualBox:~$ export PATH=$JAVA_HOME/bin:$PATH
b1809385@b1809385-VirtualBox:~$ java -version
openjdk version "1.8.0-262"
OpenJDK Runtime Environment (build 1.8.0-262-b10)
OpenJDK 64-Bit Server VM (build 25.71-b10, mixed mode)
b1809385@b1809385-VirtualBox:~$ echo $JAVA_HOME
/home/b1809385/softs/openlogic-openjdk-8u262-b10-linux-64
b1809385@b1809385-VirtualBox:~$
```

Câu 10: Thực thi dự án theo yêu cầu:

- Trong thư mục cá nhân tạo thư mục src bằng lệnh: `mkdir ~/src`
- Tạo cấu trúc thư mục src ban đầu như sau:

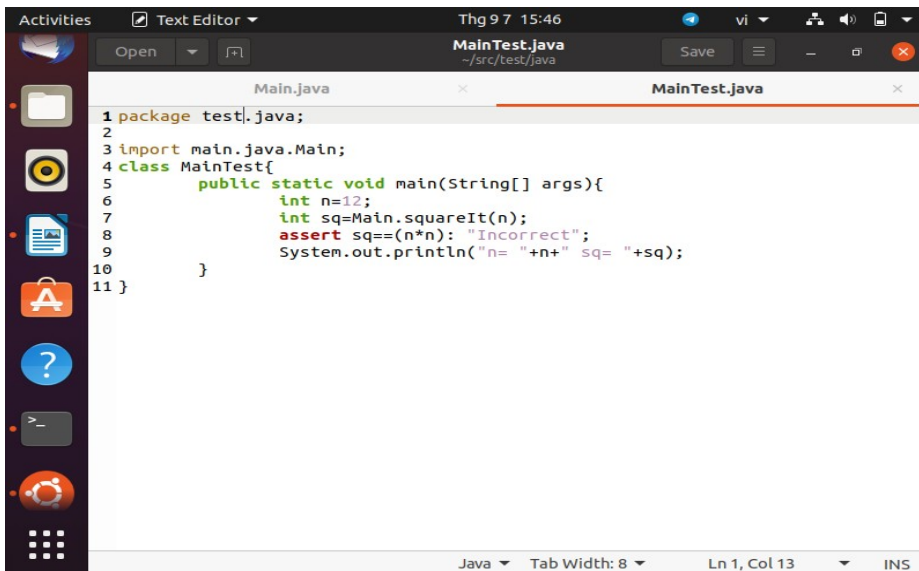
```
Activities Terminal Thg 9 7 15:52
b1809385@b1809385-VirtualBox: ~/src
b1809385@b1809385-VirtualBox:~/src$ tree
.
├── main
│   └── java
│       └── Main.java
└── test
    └── java
        └── MainTest.java

4 directories, 2 files
b1809385@b1809385-VirtualBox:~/src$
```

- Nội dung file Main.java

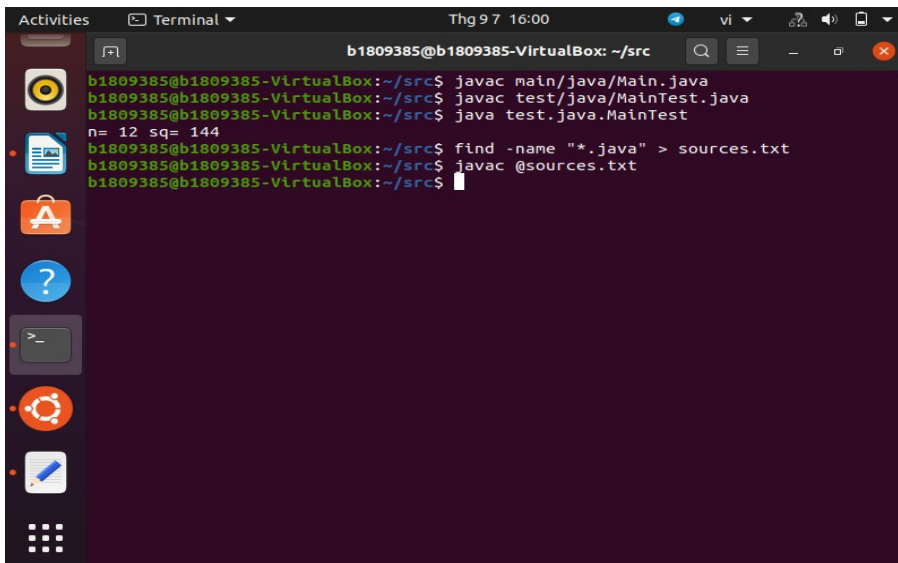
```
Activities Text Editor Thg 9 7 15:36
Main.java
~/src/main/java
1 package main.java;
2
3 import java.io.*;
4
5 public class Main{
6     public static int squareIt(int n){
7         return n*n;
8     }
9 }
```

- Nội dung file MainTest.java



```
1 package test.java;
2
3 import main.java.Main;
4 class MainTest{
5     public static void main(String[] args){
6         int n=12;
7         int sq=Main.squareIt(n);
8         assert sq==(n*n): "Incorrect";
9         System.out.println("n= "+n+" sq= "+sq);
10    }
11 }
```

- Biên dịch 2 file Main.java và MainTest.java bằng: *javac*
- Chạy file MainTest.java là file thực thi chính bằng: *java*
- *Kết quả sau khi chạy: n=12 sq=144*
- Khi muốn biên dịch nhiều file .java đồng thời, ta có thể tìm và đưa các file java vào trong 1 file, VD: sources.txt

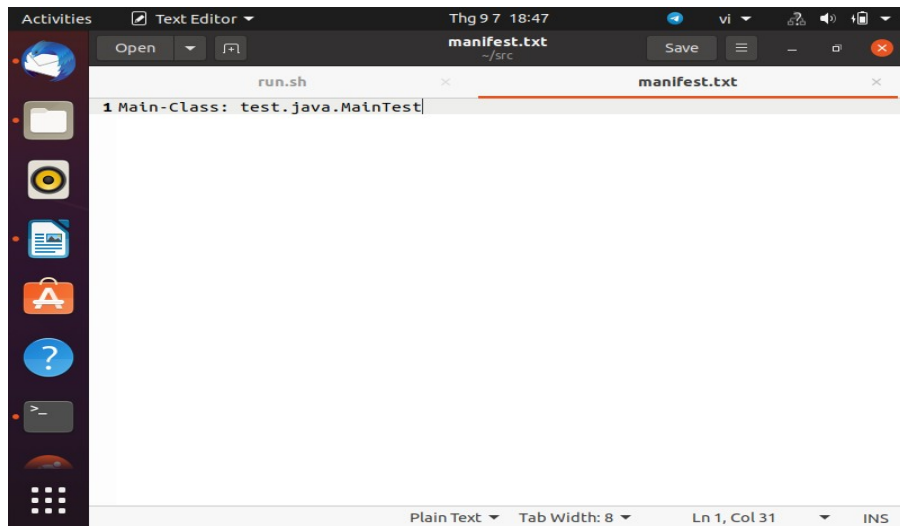


```
b1809385@b1809385-VirtualBox: ~/src
b1809385@b1809385-VirtualBox:~/src$ javac main/java/Main.java
b1809385@b1809385-VirtualBox:~/src$ javac test/java/MainTest.java
b1809385@b1809385-VirtualBox:~/src$ java test.java.MainTest
n= 12 sq= 144
b1809385@b1809385-VirtualBox:~/src$ find -name "*.java" > sources.txt
b1809385@b1809385-VirtualBox:~/src$ javac @sources.txt
b1809385@b1809385-VirtualBox:~/src$
```

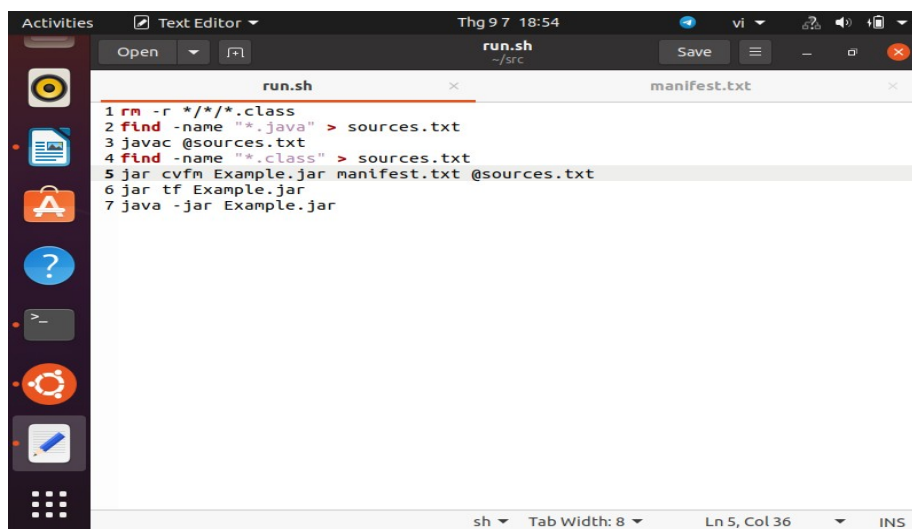
- JAR và tập lệnh shell:

- JAR là định dạng tệp gói được sử dụng để tổng hợp nhiều tệp java,...thành 1 tệp duy nhất để phân phối
- shell sẽ giúp tự động hóa, thay vì mỗi lần chạy ta phải gõ lặp đi lặp lại các lệnh, thì ta nên đưa các lệnh cần chạy vào 1 shell, và ta chỉ cần thực thi shell đó

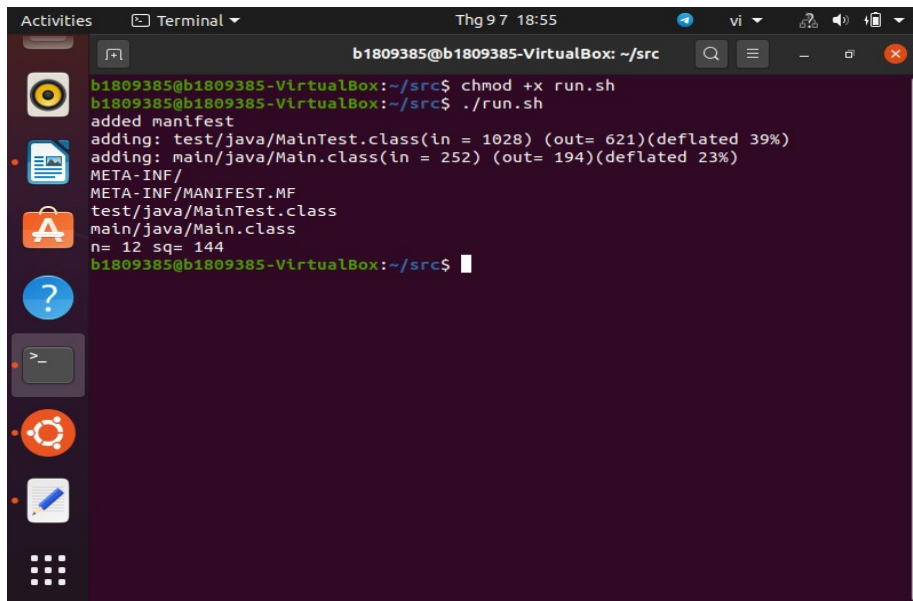
+ B1: vào src tạo file manifest.txt, gõ dòng sau để chỉ ra lớp mà tệp JAR sẽ chạy, cụ thể trong trường hợp này là MainTest:



- + B2: Trong src, tạo run.sh chứa các lệnh cần thực thi . Trong đó:
 - Lệnh dòng 1: xóa các compiled trước đó
 - Lệnh dòng 2 : tìm tất cả các file .java đưa vào sources.txt
 - Lệnh dòng 3: biên dịch file sources.txt
 - Lệnh dòng 4: tìm tất cả các file .class đưa vào sources.txt
 - Lệnh dòng 5: tạo tệp jar
 - Lệnh dòng 6: lấy nội dung tệp jar
 - Lệnh dòng 7: chạy tệp jar



- + B3: Cấp quyền để chạy run.sh bằng lệnh *chmod*. Kết quả sau khi chạy:



```
b1809385@b1809385-VirtualBox: ~/src
b1809385@b1809385-VirtualBox:~/src$ chmod +x run.sh
b1809385@b1809385-VirtualBox:~/src$ ./run.sh
added manifest
adding: test/java/MainTest.class(in = 1028) (out= 621)(deflated 39%)
adding: main/java/Main.class(in = 252) (out= 194)(deflated 23%)
META-INF/
META-INF/MANIFEST.MF
test/java/MainTest.class
main/java/Main.class
n= 12 sq= 144
b1809385@b1809385-VirtualBox:~/src$
```

---Hết---