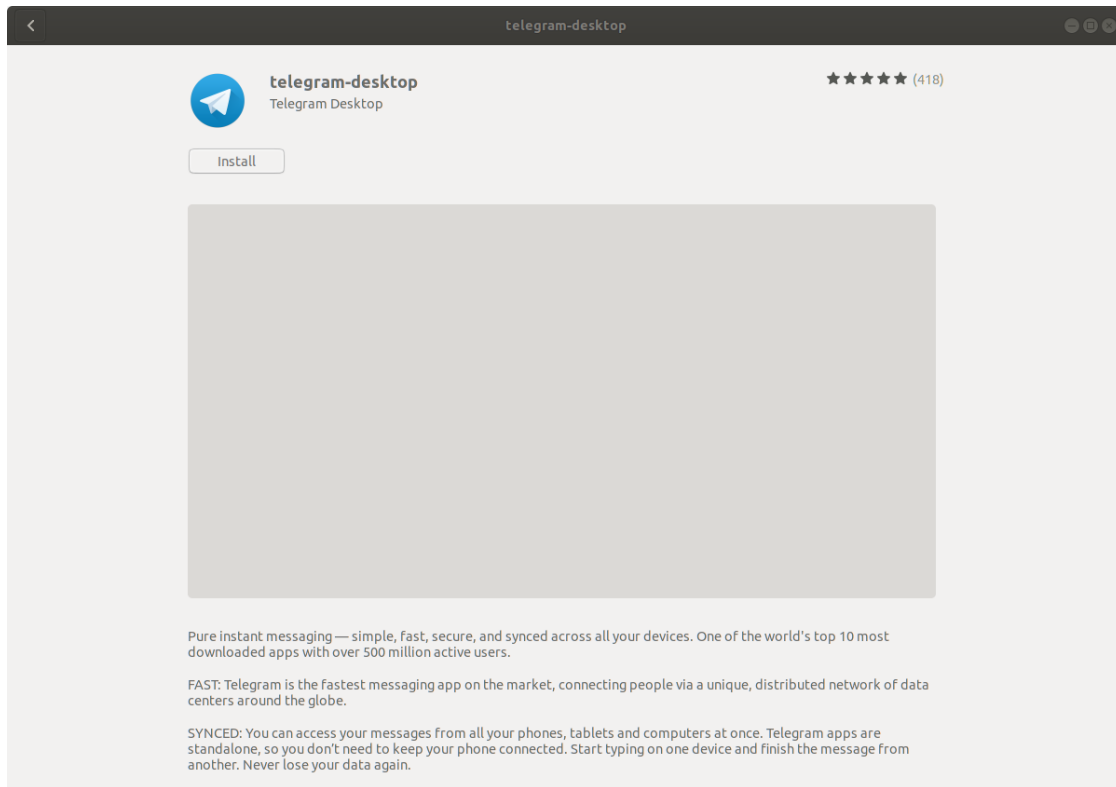


**Họ Tên:** Phạm Thị Hồng Linh  
**MSSV:** B1809365

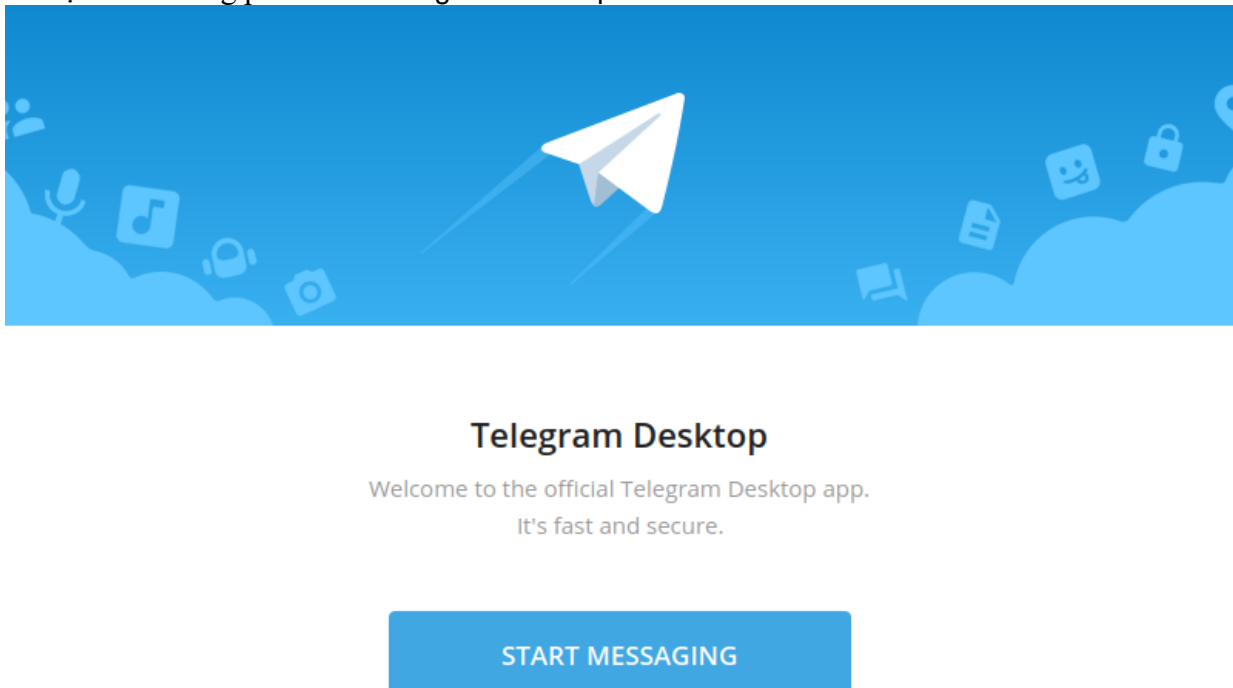
## **CHỦ ĐỀ: MÔ HÌNH MẠNG TCP/IP**

### **Chủ điểm: Cài đặt Ubuntu Desktop**

**Câu 5:** Trong Ubuntu Software: tìm kiếm **telegram-desktop** và chọn **Install** để cài



- Cài đặt thành công phần mềm telegram-desktop:



- Tiến hành cấu hình:

+ Nhập số điện thoại, họ tên để đăng ký tài khoản

**Câu 6:** Tạo thư mục softs

```
linhb1809365@linhb1809365-VirtualBox:~$ mkdir ~/softs
```

```
linhb1809365@linhb1809365-VirtualBox:~$ ls
Desktop  Downloads  Pictures  snap  Templates
Documents Music     Public   softs  Videos
```

**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**

```
linhb1809365@linhb1809365-VirtualBox:~$ java -version
Command 'java' not found, but can be installed with:

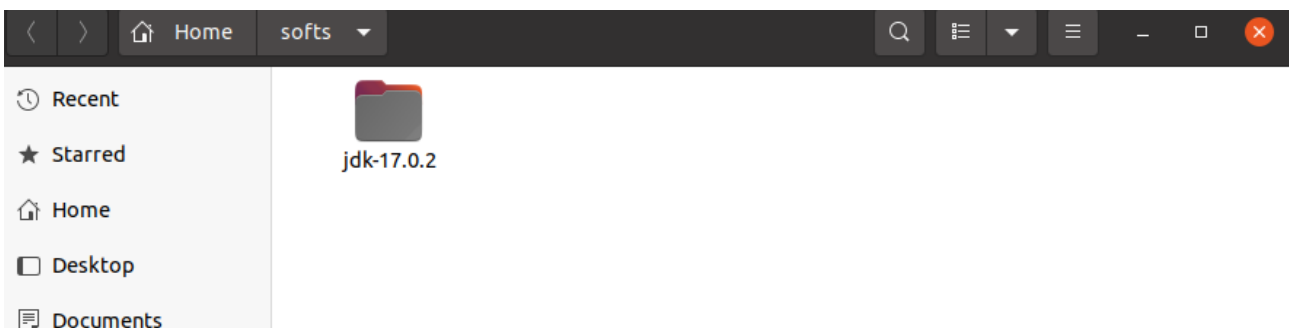
sudo apt install openjdk-11-jre-headless # version 11.0.13+8-0ubuntu1~20.04, o
r
sudo apt install default-jre             # version 2:1.11-72
sudo apt install openjdk-13-jre-headless # version 13.0.7+5-0ubuntu1~20.04
sudo apt install openjdk-16-jre-headless # version 16.0.1+9-1~20.04
sudo apt install openjdk-17-jre-headless # version 17.0.1+21-1~20.04
sudo apt install openjdk-8-jre-headless  # version 8u312-b07-0ubuntu1~20.04

linhb1809365@linhb1809365-VirtualBox:~$
```

```
linhb1809365@linhb1809365-VirtualBox:~$ hostnamectl
Static hostname: linhb1809365-VirtualBox
Icon name: computer-vm
Chassis: vm
Machine ID: a1b84b8ba9614699918a7fd427ead128
Boot ID: c947c5dfc162470a94f6152fb50c6b79
Virtualization: oracle
Operating System: Ubuntu 20.04.3 LTS
Kernel: Linux 5.13.0-28-generic
Architecture: x86_64

linhb1809365@linhb1809365-VirtualBox:~$
```

- 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 17.0.1 được cài đặt

```
linhb1809365@linhb1809365-VirtualBox:~/softs$ export JAVA_HOME="/home/linhb1809365/softs/jdk-17.0.1"
linhb1809365@linhb1809365-VirtualBox:~/softs$ export PATH=$JAVA_HOME/bin:$PATH
linhb1809365@linhb1809365-VirtualBox:~/softs$ java --version
java 17.0.1 2021-10-19 LTS
Java(TM) SE Runtime Environment (build 17.0.1+12-LTS-39)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.1+12-LTS-39, mixed mode, sharing)
linhb1809365@linhb1809365-VirtualBox:~/softs$
```

**Câu 9:** Kiểm tra giá trị biến môi trường:

Thực hiện thông qua lệnh echo \$JAVA\_HOME

```
linhb1809365@linhb1809365-VirtualBox:~$ echo $JAVA_HOME
/home/linhb1809365/softs/jdk-17.0.1
linhb1809365@linhb1809365-VirtualBox:~$
```

**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:

```
linhb1809365@linhb1809365-VirtualBox:~/src$ tree
.
├── main
│   └── java
│       └── Main.java
└── test
    └── java
        └── MainTest.java

4 directories, 2 files
linhb1809365@linhb1809365-VirtualBox:~/src$
```

- Nội dung file Main.java

```
Open  Main.java ~/src/main/java Save
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

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

- 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

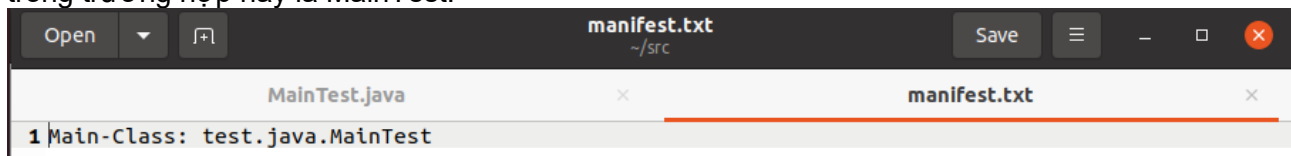
```
linhb1809365@linhb1809365-VirtualBox:~/src$ javac test/java/MainTest.java
linhb1809365@linhb1809365-VirtualBox:~/src$ javac main/java/Main.java
linhb1809365@linhb1809365-VirtualBox:~/src$ java test.java.MainTest
n= 12 sq= 144
linhb1809365@linhb1809365-VirtualBox:~/src$ find -name "*.java" > sources.txt
linhb1809365@linhb1809365-VirtualBox:~/src$ javac @sources.txt
linhb1809365@linhb1809365-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 từ trước đó

Lệnh dòng 2 : tìm tất cả các file .java đưa vào sources.txt

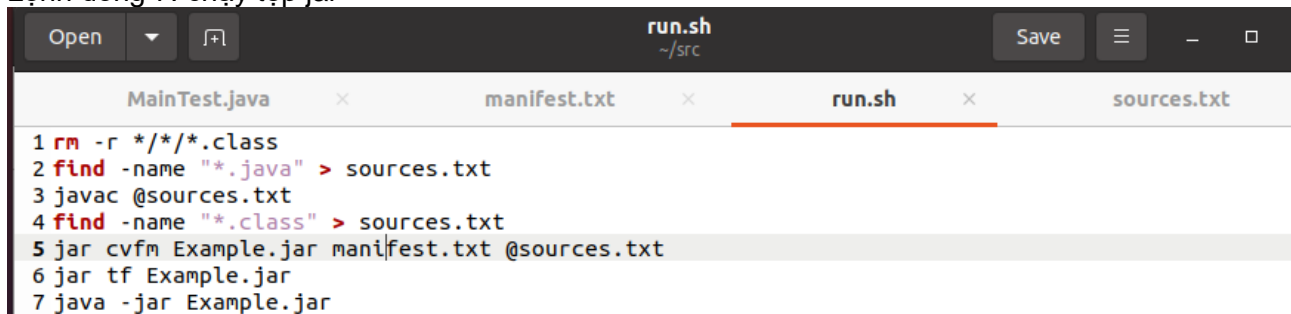
Lệnh dòng 3: biên dịch file sources.txt

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:

```
linhb1809365@linhb1809365-VirtualBox:~/src$ chmod +x run.sh
linhb1809365@linhb1809365-VirtualBox:~/src$ ./run.sh
```

```
linhb1809365@linhb1809365-VirtualBox:~/src$ ./run.sh
added manifest
adding: test/java/MainTest.class(in = 1241) (out= 710)(deflated 42%)
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
linhb1809365@linhb1809365-VirtualBox:~/src$
```