CSL1010 Project

Malware Designing

Group 29

Group Members

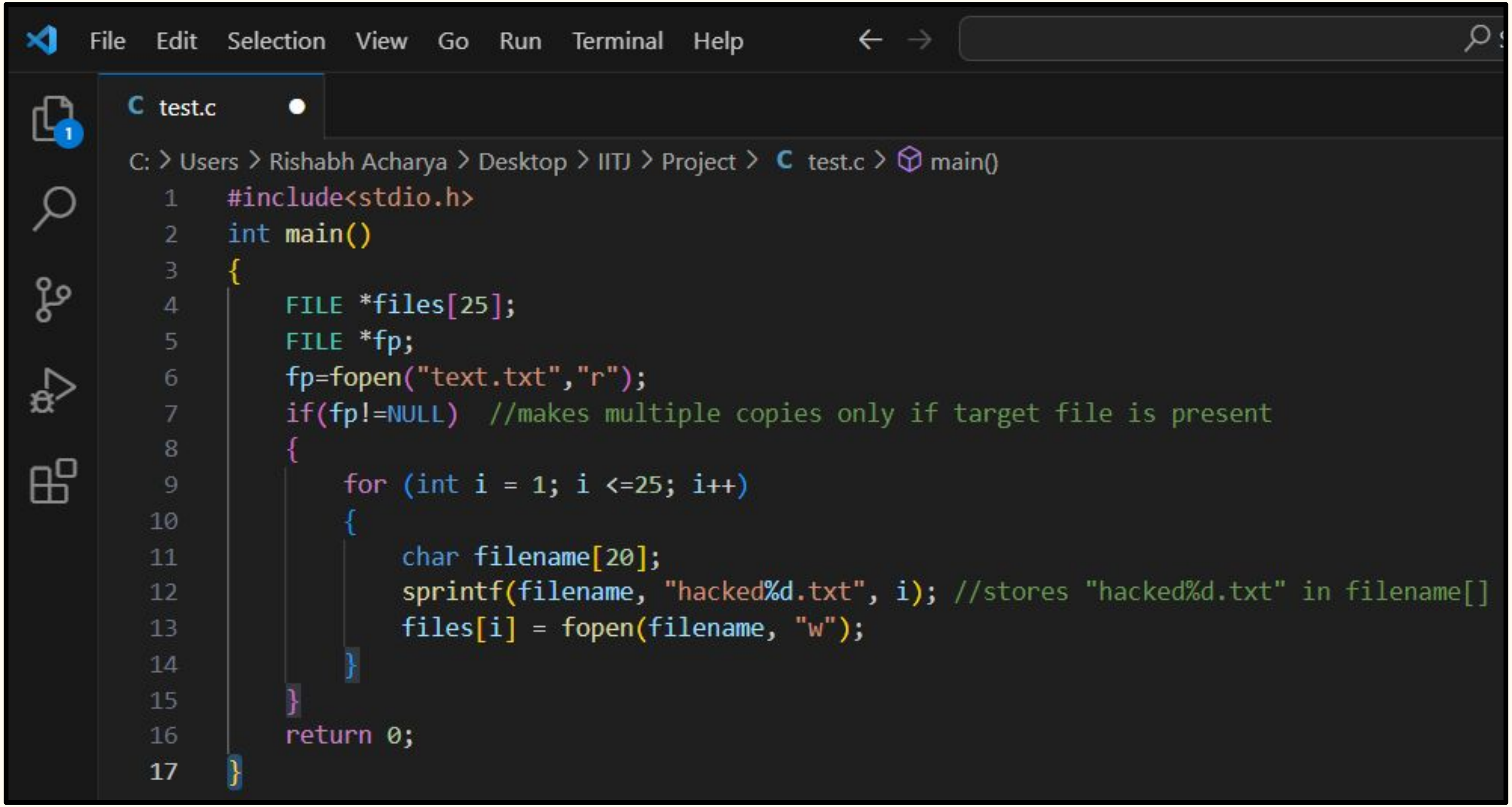
Pujit Jha (B22CI033)

Rishabh Acharya (B22CI036)

What did we try before?

We watched few youtube videos to understand how a malware works. We tried to understand how a file is hidden while it's running, and what are the potential risks of having a malware.

Our initial thought was to try and code a program that would, on clicking a specified file, make enormous number of files in the computer and proceed to make a malware with it. We did it using sprintf function. However, the idea was scraped off as nothing as such could be thought out of it. The following slide contains the basic code.


File
Edit
Selection
View
Go
Run
Terminal
Help
test.c
C: > Users > Rishabh Acharya > Desktop > IITJ > Project > test.c > main()

```c
#include<stdio.h>
int main()
{
    FILE *files[25];
    FILE *fp;
    fp=fopen("text.txt","r");
    if(fp!=NULL)  //makes multiple copies only if target file is present
    {
        for (int i = 1; i <=25; i++)
        {
            char filename[20];
            sprintf(filename, "hacked%d.txt", i); //stores "hacked%d.txt" in filename[]
            files[i] = fopen(filename, "w");
        }
    }
    return 0;
}
```

Topic that we first came up with

To design a malware that would open the camera and would keep it hidden, and on triggering a key, it would record and save it in the secondary memory of the computer. Every task would be done in the background, without letting the user know.

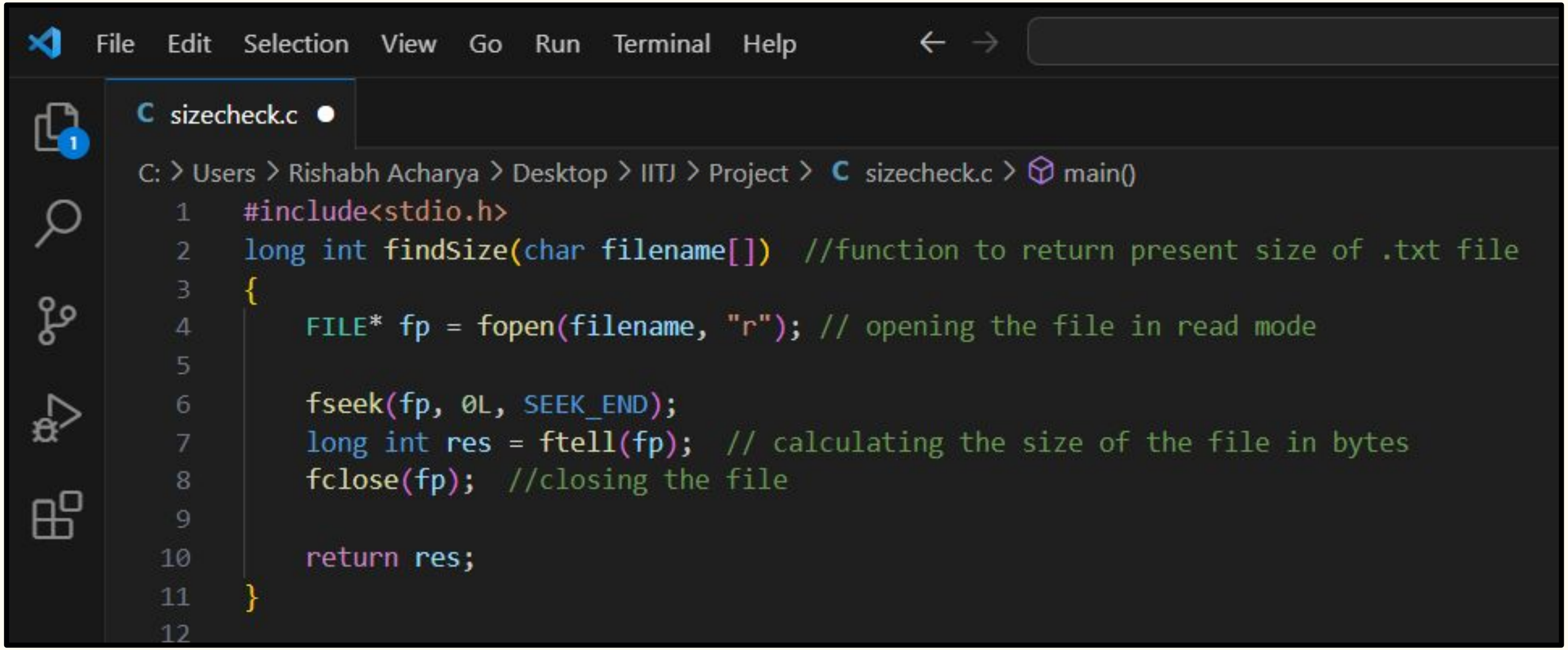
Windows Keylogger - our final topic

- It is a type of surveillance technology used to monitor and record each keystroke. Essential information such as passwords can be known through this.
- .txt file used to store each keystroke.
- Starts running on startup.

Modification we couldn’t implement

- Our idea was to check the file size everytime a new character was added.
- When the file size would have been close to the maximum permissible .txt file size, using sprintf function and file handling, we would have automatically created a new file, such as “file1.txt”, “file2.txt”, and new keystrokes would have been stored in the newly created file.
- This would have enabled the code to keep on running forever, without the problem of the .txt file running out of memory.

Basic code to find out length of file after each input

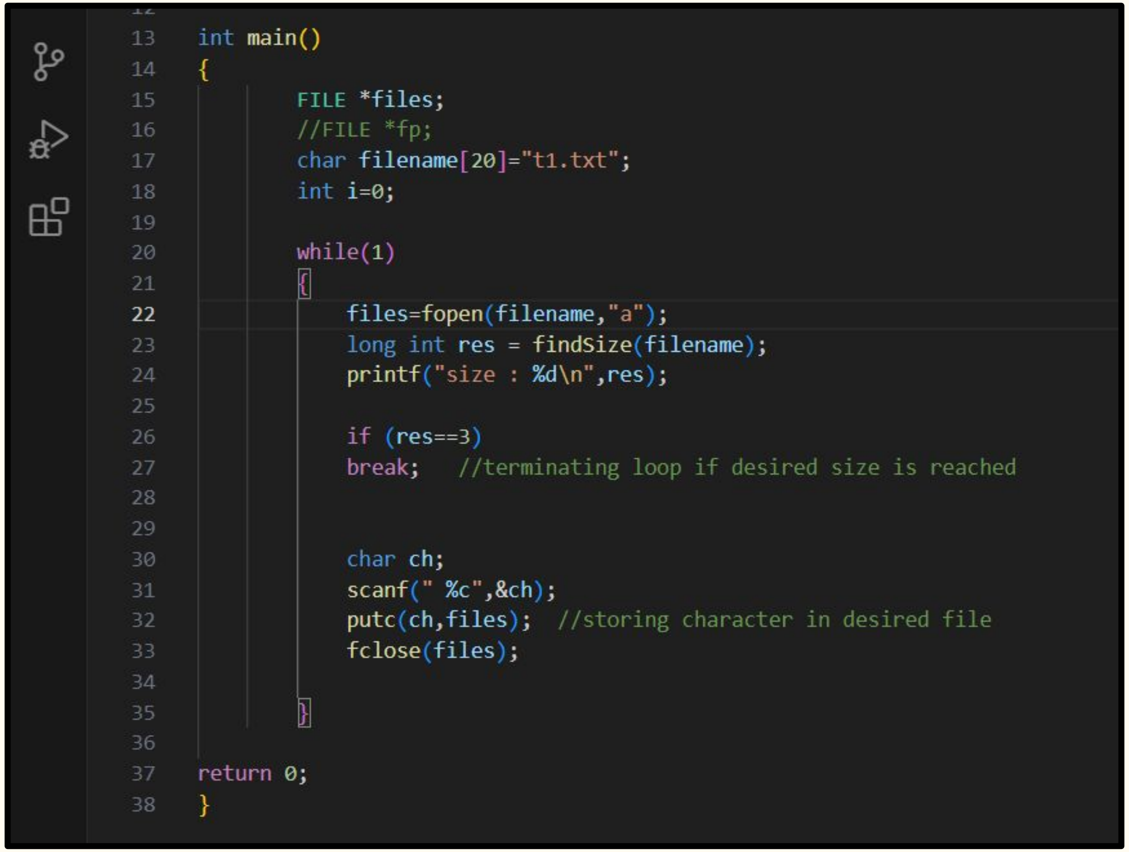

File Edit Selection View Go Run Terminal Help
sizecheck.c
C: > Users > Rishabh Acharya > Desktop > IITJ > Project > sizecheck.c > main()

```c
#include<stdio.h>
long int findSize(char filename[])  //function to return present size of .txt file
{
    FILE* fp = fopen(filename, "r"); // opening the file in read mode

    fseek(fp, 0L, SEEK_END);
    long int res = ftell(fp);  // calculating the size of the file in bytes
    fclose(fp);  //closing the file

    return res;
}
```

Basic code to find out length of file after each input



```c
int main()
{
        FILE *files;
        //FILE *fp;
        char filename[20]="t1.txt";
        int i=0;

        while(1)
        {
            files=fopen(filename,"a");
            long int res = findSize(filename);
            printf("size : %d\n",res);

            if (res==3)
            break;   //terminating loop if desired size is reached


            char ch;
            scanf(" %c",&ch);
            putc(ch,files);  //storing character in desired file
            fclose(files);

        }

return 0;
}
```

THANK YOU