GTU Department of Computer Engineering
CSE 344 - SPRING 2022
HOMEWORK 2 REPORT

Burak ÇİÇEK
1901042260

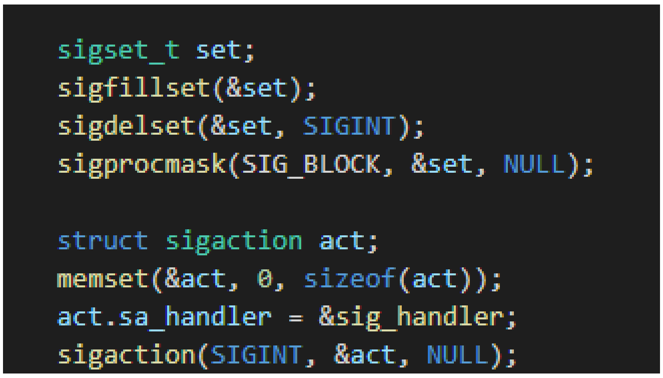
1.Design Decisions and Concept

First of all, if I need to explain the design in general. At the very beginning of the program, I do the necessary operations to mask the signal. At the same time, I provide the registration of my signal handler. Then I do the error checking of the command line arguments. I get the command line argument with the help of getopt() and process it. Then I separate the InputFilePath and the outPutFilePath with it. Then I read my input file and if my input file has less than 60 characters, I terminate the program. If it is greater than or equal to 60, I start the file read operations and create a child process every 30 characters and perform the calculations in parallel. At the end of the file reading process, I wait for the child processes to finish their processes. When all the processes are finished and all the child processes have written to the file, I read the file again. Child processes write to the file as a single line with spaces in between. While reading, I ensure the reading process with respect to these spaces. Then I find the frobenius norm by doing the necessary mathematical operations. This was the design of the parent process in general.

If we talk about the design of the child process, at the very beginning of the program, I do the necessary actions to mask the signal, and at the same time, I record the handler of my signal. Then I pass the variables I get with the help of environ to 3 different arrays x,y,z. First I find the median, then I find the variance variable, and finally I find the covariance variables and create the relevant covariance matrix. While writing to the file, I keep it locked with the help of F_SETLKW and prevent its simultaneous writing.

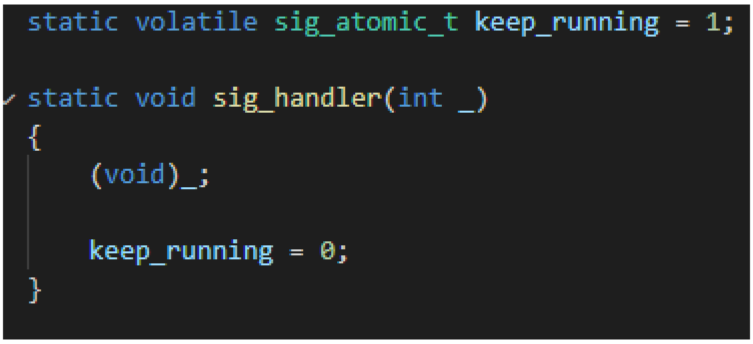
If I need to explain a little more detailed and relevant parts.

–I do the relevant signal handler and masking as follows.



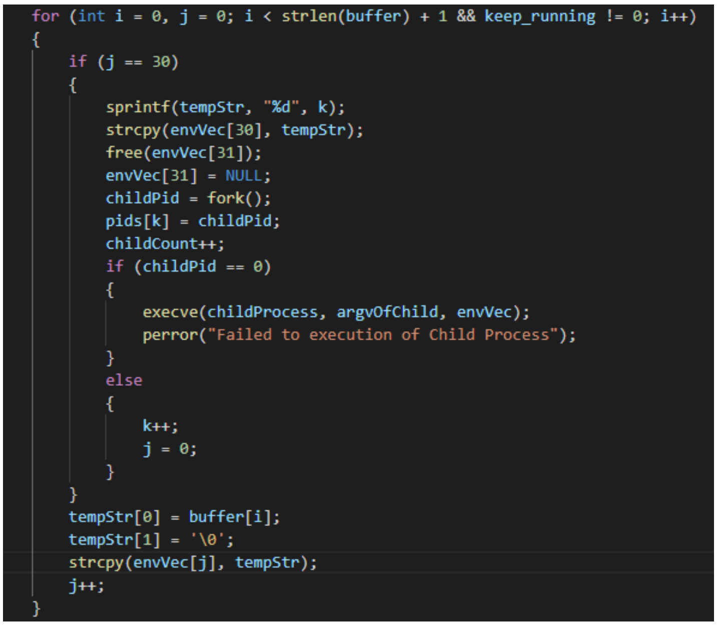
```
sigset_t set;
sigfillset(&set);
sigdelset(&set, SIGINT);
sigprocmask(SIG_BLOCK, &set, NULL);

struct sigaction act;
memset(&act, 0, sizeof(act));
act.sa_handler = &sig_handler;
sigaction(SIGINT, &act, NULL);
```

–There is not a ton of code inside the Handler function, I only have one flag argument, the operations are inside that flag block.



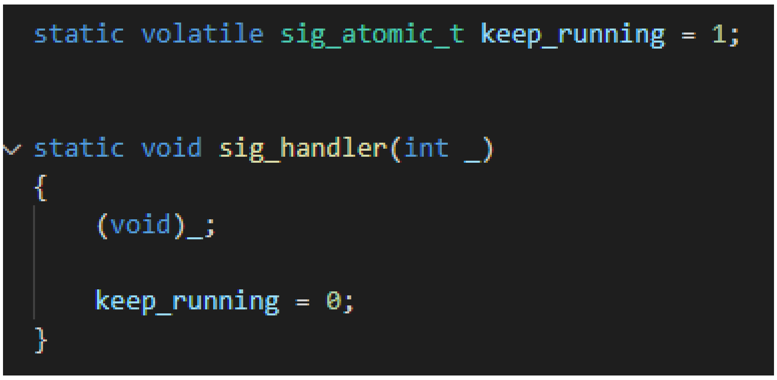
```
static volatile sig_atomic_t keep_running = 1;

static void sig_handler(int _)
{
    (void)_;

    keep_running = 0;
}
```

–I create a new child process every 30 characters and run it as a different program with execv. I pass my variables with the help of enVec.

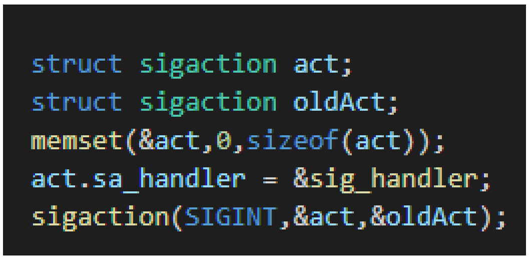


```
for (int i = 0, j = 0; i < strlen(buffer) + 1 && keep_running != 0; i++)
{
    if (j == 30)
    {
        sprintf(tempStr, "%d", k);
        strcpy(envVec[30], tempStr);
        free(envVec[31]);
        envVec[31] = NULL;
        childPid = fork();
        pids[k] = childPid;
        childCount++;
        if (childPid == 0)
        {
            execve(childProcess, argvOfChild, envVec);
            perror("Failed to execution of Child Process");
        }
        else
        {
            k++;
            j = 0;
        }
    }
    tempStr[0] = buffer[i];
    tempStr[1] = '\0';
    strcpy(envVec[j], tempStr);
    j++;
}
```

–In the child process, I first perform the masking and handler registration operations. My Handler still has the same flag logic.

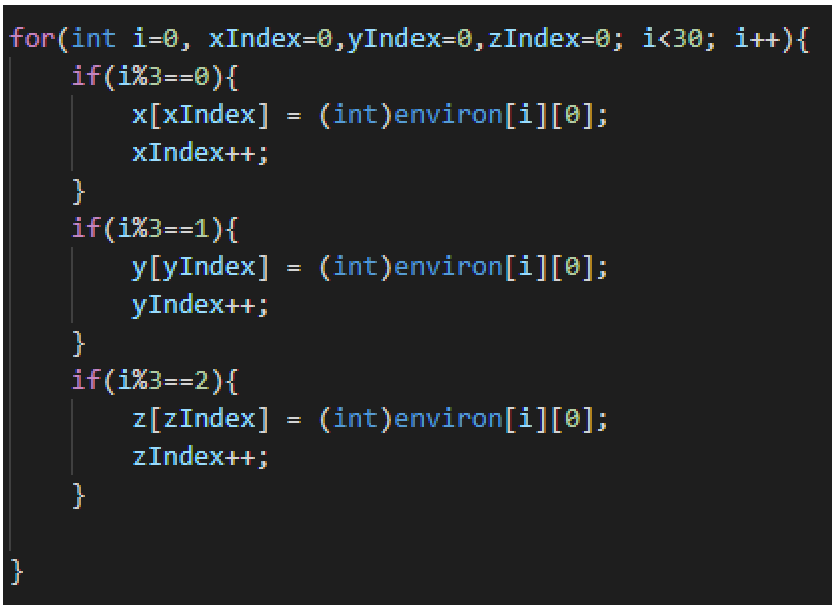


```
static volatile sig_atomic_t keep_running = 1;


static void sig_handler(int _)
{
    (void)_;

    keep_running = 0;
}
```



```
struct sigaction act;
struct sigaction oldAct;
memset(&act,0,sizeof(act));
act.sa_handler = &sig_handler;
sigaction(SIGINT,&act,&oldAct);
```

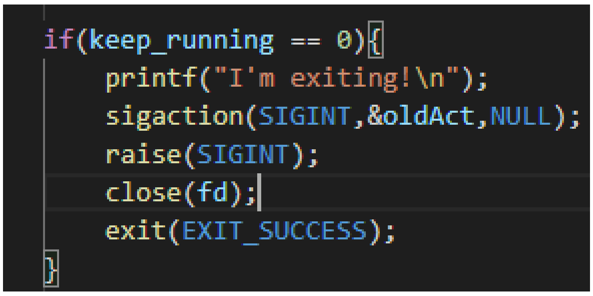
–I get my variables with the help of extern char **environ and place them in separate arrays.



```
for(int i=0, xIndex=0,yIndex=0,zIndex=0; i<30; i++){
    if(i%3==0){
        x[xIndex] = (int)environ[i][0];
        xIndex++;
    }
    if(i%3==1){
        y[yIndex] = (int)environ[i][0];
        yIndex++;
    }
    if(i%3==2){
        z[zIndex] = (int)environ[i][0];
        zIndex++;
    }

}
```

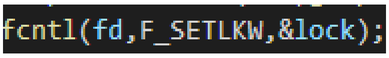
–After that calculation of median, variance operations after that I placed related variables on the CovarianceMatrix Array.

-By the way, In my Signal Handler flag block, I did my operations as the written in Lecture PDFs



```
if(keep_running == 0){
    printf("I'm exiting!\n");
    sigaction(SIGINT,&oldAct,NULL);
    raise(SIGINT);
    close(fd);
    exit(EXIT_SUCCESS);
}
```

–I perform synchronous writing between child processes with the help of F_SETLKW.



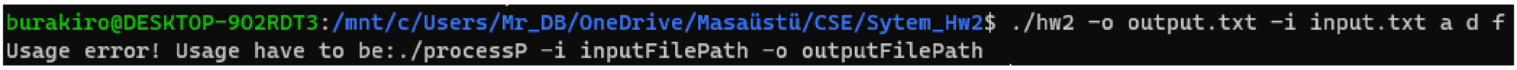
```
fcntl(fd,F_SETLKW,&lock);
```

–I delete the output file with the unlink system call in the signal handler block inside the parent process. I clear the memory and close the open file.

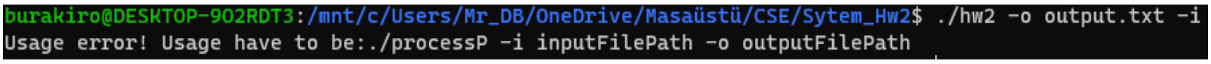
–At the end of the first loop I use while ((wpid = wait(NULL)) > 0); for the wait all Children Process' operations.

–After the information from child processes is written to the file, I read the relevant file. I save the information to the array and perform the related mathematical operations, and finally I find the result and print it on the screen.

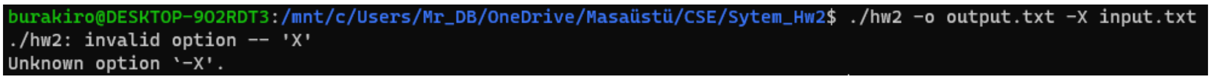
–Finally, before I put the screenshot of the tests, I achieved all the requirements stated in the assignment and I must say that there is no missing or failed feature in this homework. I did my code and tests on Dual Boot Linux, but I pulled my ss's from WSL.



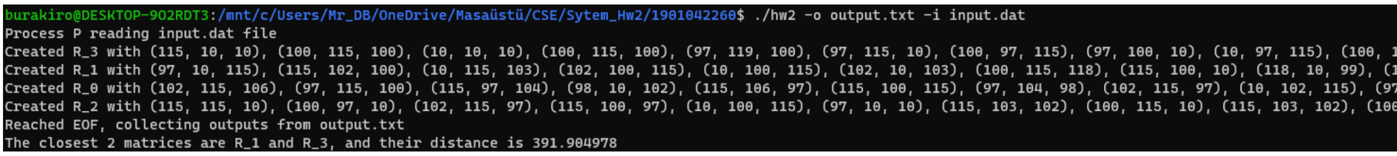
```
burakiro@DESKTOP-902RDT3:/mnt/c/Users/Mr_DB/OneDrive/Masaüstü/CSE/Sytem_Hw2$ ./hw2 -o output.txt -i input.txt a d f
Usage error! Usage have to be:./processP -i inputFilePath -o outputFilePath
```



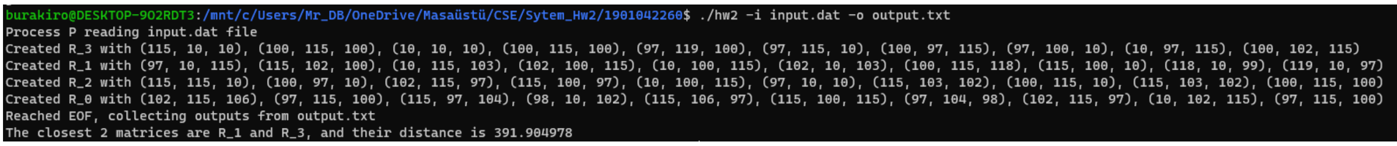
```
burakiro@DESKTOP-902RDT3:/mnt/c/Users/Mr_DB/OneDrive/Masaüstü/CSE/Sytem_Hw2$ ./hw2 -o output.txt -i
Usage error! Usage have to be:./processP -i inputFilePath -o outputFilePath
```



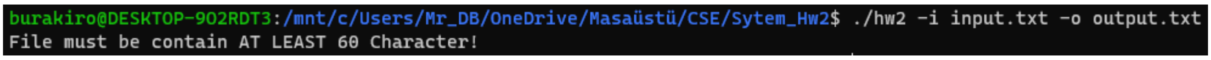
```
burakiro@DESKTOP-902RDT3:/mnt/c/Users/Mr_DB/OneDrive/Masaüstü/CSE/Sytem_Hw2$ ./hw2 -o output.txt -X input.txt
./hw2: invalid option -- 'X'
Unknown option `-X'.
```



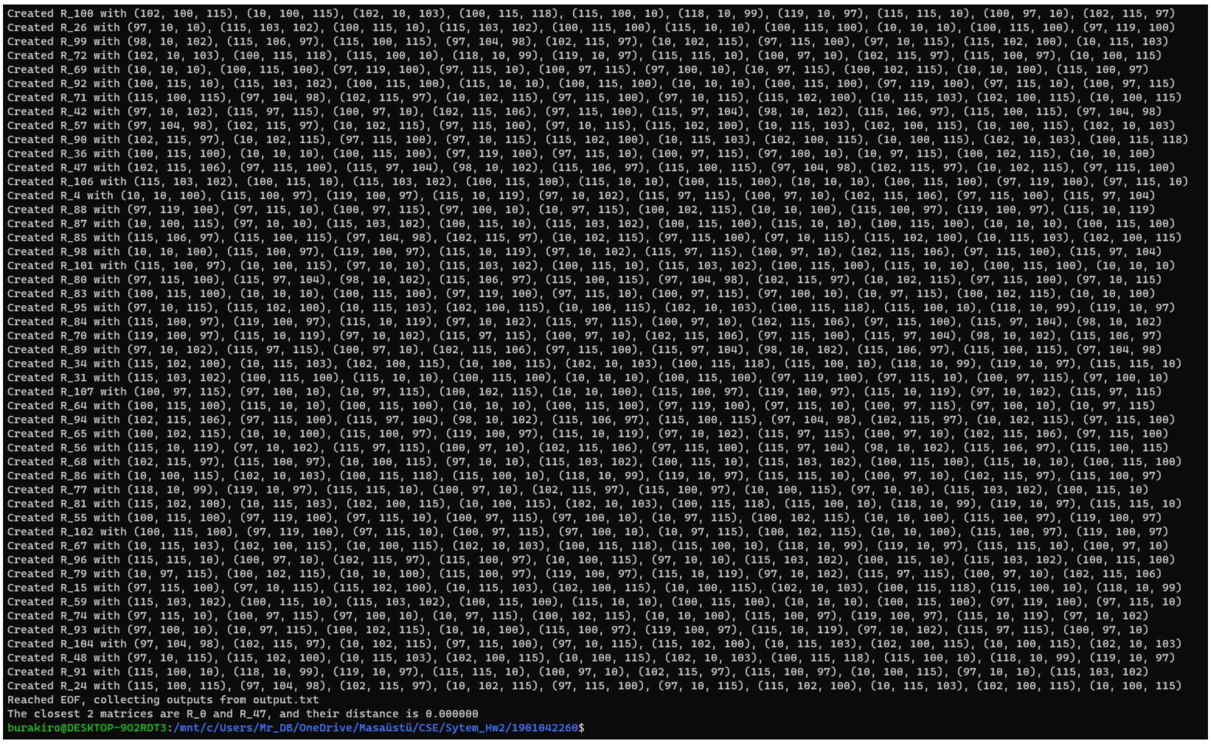
```
burakiro@DESKTOP-902RDT3:/mnt/c/Users/Mr_DB/OneDrive/Masaüstü/CSE/Sytem_Hw2/1901042260$ ./hw2 -o output.txt -i input.dat
Process P reading input.dat file
Created R_3 with (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 1
Created R_1 with (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (1
Created R_0 with (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97), (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97
Created R_2 with (115, 115, 10), (100, 97, 10), (102, 115, 97), (115, 100, 97), (10, 100, 115), (97, 10, 10), (115, 103, 102), (100, 115, 10), (115, 103, 102), (100
Reached EOF, collecting outputs from output.txt
The closest 2 matrices are R_1 and R_3, and their distance is 391.904978
```



```
burakiro@DESKTOP-9O2RDT3:/mnt/c/Users/Mr_DB/OneDrive/Masaüstü/CSE/Sytem_Hw2/1901042260$ ./hw2 -i input.dat -o output.txt
Process P reading input.dat file
Created R_3 with (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115)
Created R_1 with (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97)
Created R_2 with (115, 115, 10), (100, 97, 10), (102, 115, 97), (115, 100, 97), (10, 100, 115), (97, 10, 10), (115, 103, 102), (100, 115, 10), (115, 103, 102), (100, 115, 100)
Created R_0 with (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97), (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100)
Reached EOF, collecting outputs from output.txt
The closest 2 matrices are R_1 and R_3, and their distance is 391.904978
```

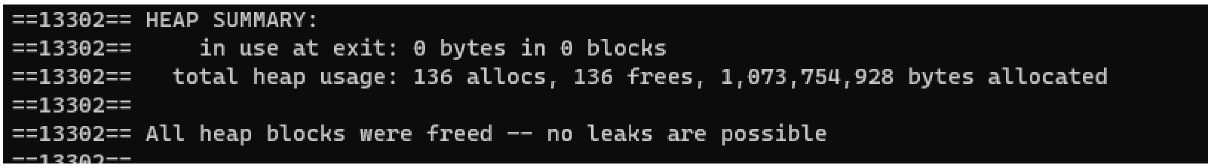


```
burakiro@DESKTOP-9O2RDT3:/mnt/c/Users/Mr_DB/OneDrive/Masaüstü/CSE/Sytem_Hw2$ ./hw2 -i input.txt -o output.txt
File must be contain AT LEAST 60 Character!
```

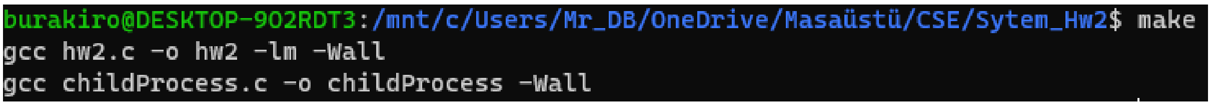


```
Created R_100 with (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97), (115, 115, 10), (100, 97, 10), (102, 115, 97)
Created R_26 with (97, 10, 10), (115, 103, 102), (100, 115, 10), (115, 103, 102), (100, 115, 100), (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100)
Created R_99 with (98, 10, 102), (115, 106, 97), (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100), (97, 10, 115), (115, 102, 100), (10, 115, 103)
Created R_72 with (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97), (115, 115, 10), (100, 97, 10), (102, 115, 97), (115, 100, 97), (10, 100, 115)
Created R_69 with (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100), (115, 100, 97)
Created R_92 with (100, 115, 10), (115, 103, 102), (100, 115, 100), (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115)
Created R_71 with (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100), (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115)
Created R_42 with (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97), (115, 100, 115), (97, 104, 98)
Created R_57 with (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100), (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103)
Created R_90 with (102, 115, 97), (10, 102, 115), (97, 115, 100), (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118)
Created R_36 with (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100)
Created R_47 with (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97), (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100)
Created R_106 with (115, 103, 102), (100, 115, 10), (115, 103, 102), (100, 115, 100), (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10)
Created R_4 with (10, 10, 100), (115, 100, 97), (119, 100, 97), (115, 10, 119), (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106), (97, 115, 100), (115, 97, 104)
Created R_88 with (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100), (115, 100, 97), (119, 100, 97), (115, 10, 119)
Created R_87 with (10, 100, 115), (97, 10, 10), (115, 103, 102), (100, 115, 10), (115, 103, 102), (100, 115, 100), (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100)
Created R_85 with (115, 106, 97), (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100), (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115)
Created R_98 with (10, 10, 100), (115, 100, 97), (119, 100, 97), (115, 10, 119), (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106), (97, 115, 100), (115, 97, 104)
Created R_101 with (115, 100, 97), (10, 100, 115), (97, 10, 10), (115, 103, 102), (100, 115, 10), (115, 103, 102), (100, 115, 100), (115, 10, 10), (100, 115, 100), (10, 10, 10)
Created R_80 with (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97), (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100), (97, 10, 115)
Created R_83 with (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100)
Created R_95 with (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97)
Created R_84 with (115, 100, 97), (119, 100, 97), (115, 10, 119), (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102)
Created R_70 with (119, 100, 97), (115, 10, 119), (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97)
Created R_89 with (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97), (115, 100, 115), (97, 104, 98)
Created R_34 with (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97), (115, 115, 10)
Created R_31 with (115, 103, 102), (100, 115, 100), (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10)
Created R_107 with (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100), (115, 100, 97), (119, 100, 97), (115, 10, 119), (97, 10, 102), (115, 97, 115)
Created R_64 with (100, 115, 100), (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115)
Created R_94 with (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97), (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100)
Created R_65 with (100, 102, 115), (10, 10, 100), (115, 100, 97), (119, 100, 97), (115, 10, 119), (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106), (97, 115, 100)
Created R_56 with (115, 10, 119), (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106), (97, 115, 100), (115, 97, 104), (98, 10, 102), (115, 106, 97), (115, 100, 115)
Created R_68 with (102, 115, 97), (115, 100, 97), (10, 100, 115), (97, 10, 10), (115, 103, 102), (100, 115, 10), (115, 103, 102), (100, 115, 100), (115, 10, 10), (100, 115, 100)
Created R_86 with (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97), (115, 115, 10), (100, 97, 10), (102, 115, 97), (115, 100, 97)
Created R_77 with (118, 10, 99), (119, 10, 97), (115, 115, 10), (100, 97, 10), (102, 115, 97), (115, 100, 97), (10, 100, 115), (97, 10, 10), (115, 103, 102), (100, 115, 10)
Created R_81 with (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97), (115, 115, 10)
Created R_55 with (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100), (115, 100, 97), (119, 100, 97)
Created R_102 with (100, 115, 100), (97, 119, 100), (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100), (115, 100, 97), (119, 100, 97)
Created R_67 with (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97), (115, 115, 10), (100, 97, 10)
Created R_96 with (115, 115, 10), (100, 97, 10), (102, 115, 97), (115, 100, 97), (10, 100, 115), (97, 10, 10), (115, 103, 102), (100, 115, 10), (115, 103, 102), (100, 115, 100)
Created R_79 with (10, 97, 115), (100, 102, 115), (10, 10, 100), (115, 100, 97), (119, 100, 97), (115, 10, 119), (97, 10, 102), (115, 97, 115), (100, 97, 10), (102, 115, 106)
Created R_15 with (97, 115, 100), (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99)
Created R_59 with (115, 103, 102), (100, 115, 10), (115, 103, 102), (100, 115, 100), (115, 10, 10), (100, 115, 100), (10, 10, 10), (100, 115, 100), (97, 119, 100), (97, 115, 10)
Created R_74 with (97, 115, 10), (100, 97, 115), (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100), (115, 100, 97), (119, 100, 97), (115, 10, 119), (97, 10, 102)
Created R_93 with (97, 100, 10), (10, 97, 115), (100, 102, 115), (10, 10, 100), (115, 100, 97), (119, 100, 97), (115, 10, 119), (97, 10, 102), (115, 97, 115), (100, 97, 10)
Created R_104 with (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100), (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103)
Created R_48 with (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115), (102, 10, 103), (100, 115, 118), (115, 100, 10), (118, 10, 99), (119, 10, 97)
Created R_91 with (115, 100, 10), (118, 10, 99), (119, 10, 97), (115, 115, 10), (100, 97, 10), (102, 115, 97), (115, 100, 97), (10, 100, 115), (97, 10, 10), (115, 103, 102)
Created R_24 with (115, 100, 115), (97, 104, 98), (102, 115, 97), (10, 102, 115), (97, 115, 100), (97, 10, 115), (115, 102, 100), (10, 115, 103), (102, 100, 115), (10, 100, 115)
Reached EOF, collecting outputs from output.txt
The closest 2 matrices are R_0 and R_47, and their distance is 0.000000
burakiro@DESKTOP-9O2RDT3:/mnt/c/Users/Mr_DB/OneDrive/Masaüstü/CSE/Sytem_Hw2/1901042260$
```

(Duplicated Long Input.txt)



```
==13302== HEAP SUMMARY:
==13302==     in use at exit: 0 bytes in 0 blocks
==13302==   total heap usage: 136 allocs, 136 frees, 1,073,754,928 bytes allocated
==13302==
==13302== All heap blocks were freed -- no leaks are possible
==13302==
```



```
burakiro@DESKTOP-9O2RDT3:/mnt/c/Users/Mr_DB/OneDrive/Masaüstü/CSE/Sytem_Hw2$ make
gcc hw2.c -o hw2 -lm -Wall
gcc childProcess.c -o childProcess -Wall
```