CSE 344
SYSTEMS PROGRAMMING

HOMEWORK 02

REPORT

ÖMER ÇEVİK
161044004

1. <u>Part 1</u>

In that part, program.c starts to read inputPath.txt as parent process **P1**. At the same time fork() syscall creates a child process which runs the **P2**.

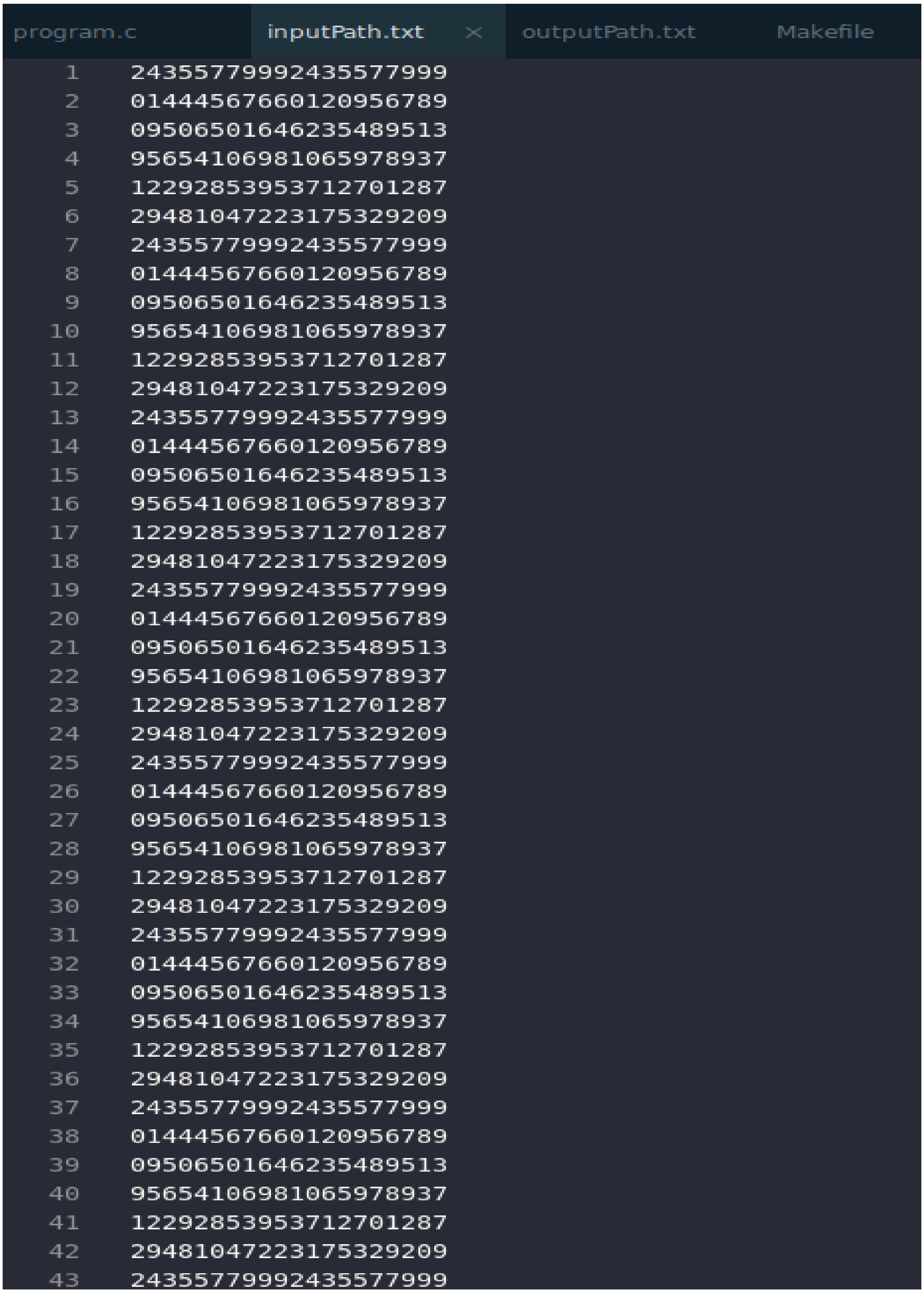
P1 process reads input file line by line and parses to the ASCII values of coordinates x and y. For each line it evaluates the **least squares method** in critical section that no signals can interrupts using **sigprocmask()** syscall. There are two signal handlers which handles **SIGTERM** signal to close files and remove input and temp files. The parent process communicates with child process which is P2, using **kill()** syscall and sends **SIGUSR2** signal. To wait the child process **sigsuspend()** syscall is used. While evaluating the result of least squares method, it writes to temp file which is created via **mkstemp** syscall. After all these success steps, P1 prints to screen that how many bytes that it read, how many lines that it read and which signals handled.

While P1 is writing in temp file, **P2** is the child process of P1 creates an outputPath.txt file to write the contents to it and starts to read temp file. For each line that it read is deleted after parsing operations. Then parsed values are evaluated by error calculator functions which are blocked to get interrupt by any signals using **sigprocmask()** system call. P2 is communicated with P1 using **SIGUSR1** signal with **kill()** system call. Each errors are written to outputPath.txt line by line as appending temp file's content. After the succeed writing operations and reading operations, P2 prints to screen index of each line, MAE, MSE and RMSE errors. After all errors' prints, it prints to screen the MAE, MSE and RMSE errors' means. Then prints to screen the standart deviations of that MAE, MSE and RMSE errors.

Signal handler functions are handled using struct of sigaction and **sigaction()** syscall. If **SIGTERM** signal becomes in execution time, then signal handler closes all opened files and removes the input and temp files. After the success of each lines are printed to temp file and read and deleted back to temp file is deleted end of the P2. All files are closed after operations clearly. Input file is deleted after P2 finishes its job.

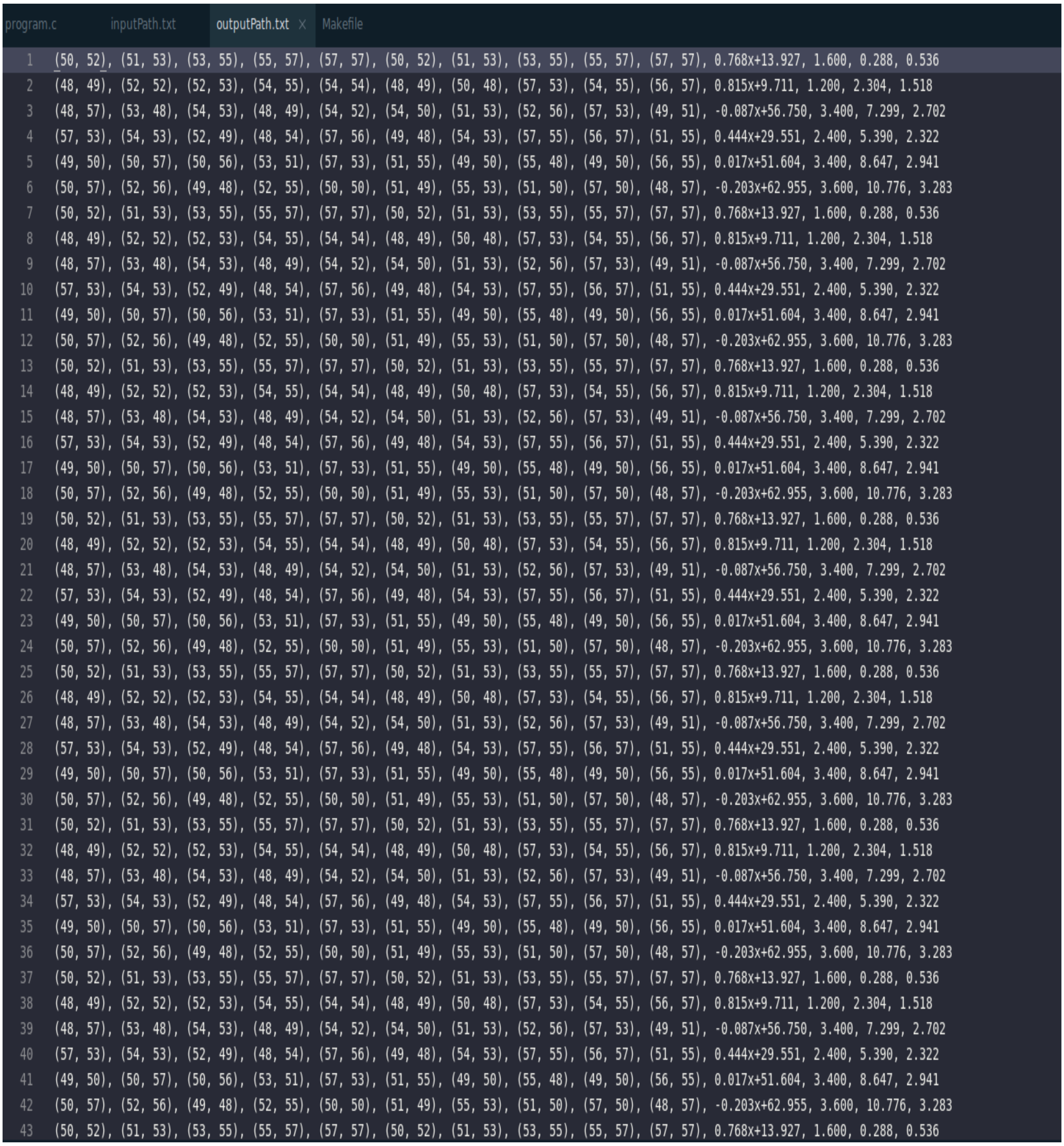
Compiler Result:

inputPath.txt:



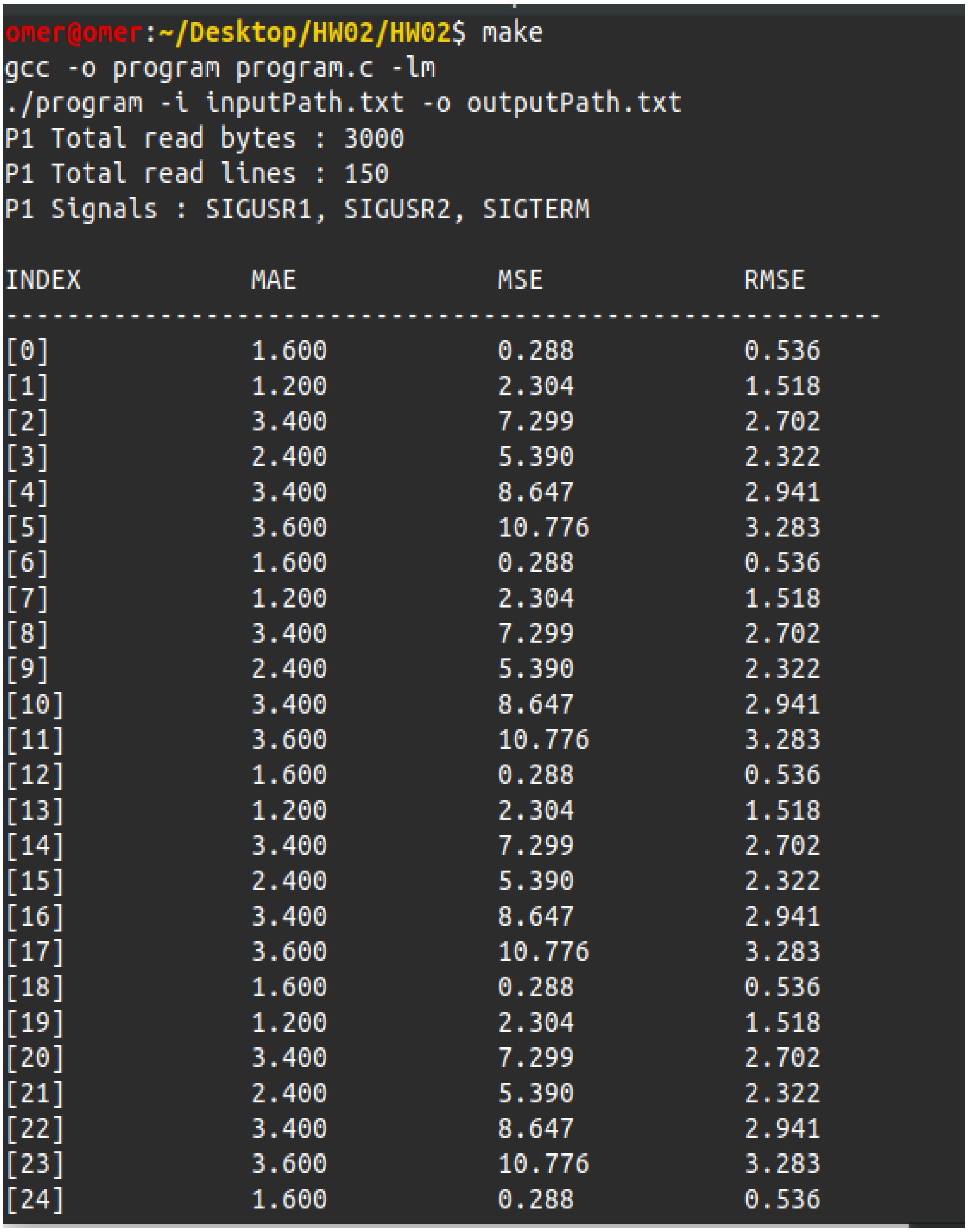
```
24355779992435577999
01444567660120956789
09506501646235489513
95654106981065978937
12292853953712701287
29481047223175329209
24355779992435577999
01444567660120956789
09506501646235489513
95654106981065978937
12292853953712701287
29481047223175329209
24355779992435577999
01444567660120956789
09506501646235489513
95654106981065978937
12292853953712701287
29481047223175329209
24355779992435577999
01444567660120956789
09506501646235489513
95654106981065978937
12292853953712701287
29481047223175329209
24355779992435577999
01444567660120956789
09506501646235489513
95654106981065978937
12292853953712701287
29481047223175329209
24355779992435577999
01444567660120956789
09506501646235489513
95654106981065978937
12292853953712701287
29481047223175329209
24355779992435577999
01444567660120956789
09506501646235489513
95654106981065978937
12292853953712701287
29481047223175329209
24355779992435577999
```

<u>outputPath.txt:</u>

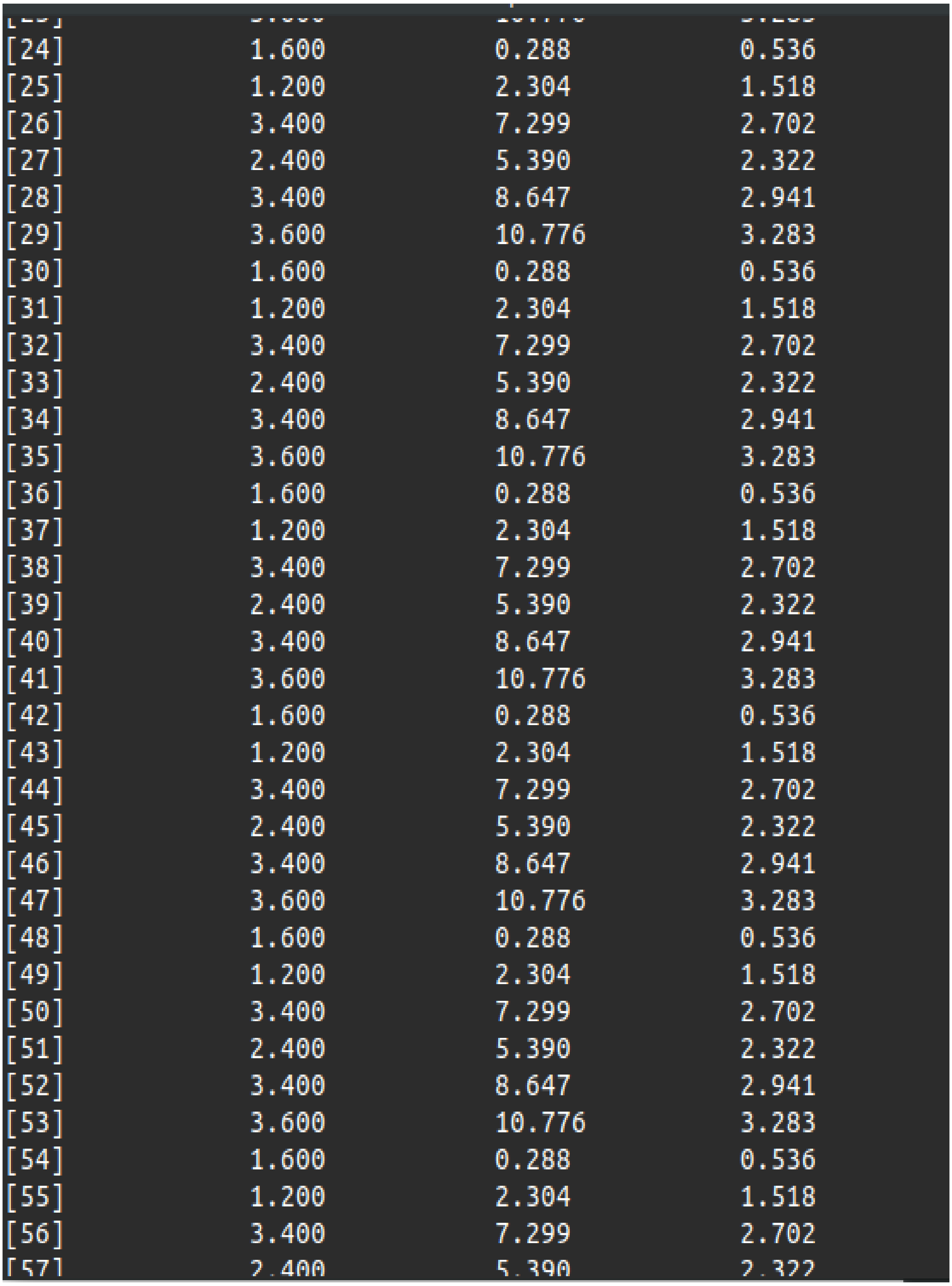


```
(50, 52), (51, 53), (53, 55), (55, 57), (57, 57), (50, 52), (51, 53), (53, 55), (55, 57), (57, 57), 0.768x+13.927, 1.600, 0.288, 0.536
(48, 49), (52, 52), (52, 53), (54, 55), (54, 54), (48, 49), (50, 48), (57, 53), (54, 55), (56, 57), 0.815x+9.711, 1.200, 2.304, 1.518
(48, 57), (53, 48), (54, 53), (48, 49), (54, 52), (54, 50), (51, 53), (52, 56), (57, 53), (49, 51), -0.087x+56.750, 3.400, 7.299, 2.702
(57, 53), (54, 53), (52, 49), (48, 54), (57, 56), (49, 48), (54, 53), (57, 55), (56, 57), (51, 55), 0.444x+29.551, 2.400, 5.390, 2.322
(49, 50), (50, 57), (50, 56), (53, 51), (57, 53), (51, 55), (49, 50), (55, 48), (49, 50), (56, 55), 0.017x+51.604, 3.400, 8.647, 2.941
(50, 57), (52, 56), (49, 48), (52, 55), (50, 50), (51, 49), (55, 53), (51, 50), (57, 50), (48, 57), -0.203x+62.955, 3.600, 10.776, 3.283
(50, 52), (51, 53), (53, 55), (55, 57), (57, 57), (50, 52), (51, 53), (53, 55), (55, 57), (57, 57), 0.768x+13.927, 1.600, 0.288, 0.536
(48, 49), (52, 52), (52, 53), (54, 55), (54, 54), (48, 49), (50, 48), (57, 53), (54, 55), (56, 57), 0.815x+9.711, 1.200, 2.304, 1.518
(48, 57), (53, 48), (54, 53), (48, 49), (54, 52), (54, 50), (51, 53), (52, 56), (57, 53), (49, 51), -0.087x+56.750, 3.400, 7.299, 2.702
(57, 53), (54, 53), (52, 49), (48, 54), (57, 56), (49, 48), (54, 53), (57, 55), (56, 57), (51, 55), 0.444x+29.551, 2.400, 5.390, 2.322
(49, 50), (50, 57), (50, 56), (53, 51), (57, 53), (51, 55), (49, 50), (55, 48), (49, 50), (56, 55), 0.017x+51.604, 3.400, 8.647, 2.941
(50, 57), (52, 56), (49, 48), (52, 55), (50, 50), (51, 49), (55, 53), (51, 50), (57, 50), (48, 57), -0.203x+62.955, 3.600, 10.776, 3.283
(50, 52), (51, 53), (53, 55), (55, 57), (57, 57), (50, 52), (51, 53), (53, 55), (55, 57), (57, 57), 0.768x+13.927, 1.600, 0.288, 0.536
(48, 49), (52, 52), (52, 53), (54, 55), (54, 54), (48, 49), (50, 48), (57, 53), (54, 55), (56, 57), 0.815x+9.711, 1.200, 2.304, 1.518
(48, 57), (53, 48), (54, 53), (48, 49), (54, 52), (54, 50), (51, 53), (52, 56), (57, 53), (49, 51), -0.087x+56.750, 3.400, 7.299, 2.702
(57, 53), (54, 53), (52, 49), (48, 54), (57, 56), (49, 48), (54, 53), (57, 55), (56, 57), (51, 55), 0.444x+29.551, 2.400, 5.390, 2.322
(49, 50), (50, 57), (50, 56), (53, 51), (57, 53), (51, 55), (49, 50), (55, 48), (49, 50), (56, 55), 0.017x+51.604, 3.400, 8.647, 2.941
(50, 57), (52, 56), (49, 48), (52, 55), (50, 50), (51, 49), (55, 53), (51, 50), (57, 50), (48, 57), -0.203x+62.955, 3.600, 10.776, 3.283
(50, 52), (51, 53), (53, 55), (55, 57), (57, 57), (50, 52), (51, 53), (53, 55), (55, 57), (57, 57), 0.768x+13.927, 1.600, 0.288, 0.536
(48, 49), (52, 52), (52, 53), (54, 55), (54, 54), (48, 49), (50, 48), (57, 53), (54, 55), (56, 57), 0.815x+9.711, 1.200, 2.304, 1.518
(48, 57), (53, 48), (54, 53), (48, 49), (54, 52), (54, 50), (51, 53), (52, 56), (57, 53), (49, 51), -0.087x+56.750, 3.400, 7.299, 2.702
(57, 53), (54, 53), (52, 49), (48, 54), (57, 56), (49, 48), (54, 53), (57, 55), (56, 57), (51, 55), 0.444x+29.551, 2.400, 5.390, 2.322
(49, 50), (50, 57), (50, 56), (53, 51), (57, 53), (51, 55), (49, 50), (55, 48), (49, 50), (56, 55), 0.017x+51.604, 3.400, 8.647, 2.941
(50, 57), (52, 56), (49, 48), (52, 55), (50, 50), (51, 49), (55, 53), (51, 50), (57, 50), (48, 57), -0.203x+62.955, 3.600, 10.776, 3.283
(50, 52), (51, 53), (53, 55), (55, 57), (57, 57), (50, 52), (51, 53), (53, 55), (55, 57), (57, 57), 0.768x+13.927, 1.600, 0.288, 0.536
(48, 49), (52, 52), (52, 53), (54, 55), (54, 54), (48, 49), (50, 48), (57, 53), (54, 55), (56, 57), 0.815x+9.711, 1.200, 2.304, 1.518
(48, 57), (53, 48), (54, 53), (48, 49), (54, 52), (54, 50), (51, 53), (52, 56), (57, 53), (49, 51), -0.087x+56.750, 3.400, 7.299, 2.702
(57, 53), (54, 53), (52, 49), (48, 54), (57, 56), (49, 48), (54, 53), (57, 55), (56, 57), (51, 55), 0.444x+29.551, 2.400, 5.390, 2.322
(49, 50), (50, 57), (50, 56), (53, 51), (57, 53), (51, 55), (49, 50), (55, 48), (49, 50), (56, 55), 0.017x+51.604, 3.400, 8.647, 2.941
(50, 57), (52, 56), (49, 48), (52, 55), (50, 50), (51, 49), (55, 53), (51, 50), (57, 50), (48, 57), -0.203x+62.955, 3.600, 10.776, 3.283
(50, 52), (51, 53), (53, 55), (55, 57), (57, 57), (50, 52), (51, 53), (53, 55), (55, 57), (57, 57), 0.768x+13.927, 1.600, 0.288, 0.536
(48, 49), (52, 52), (52, 53), (54, 55), (54, 54), (48, 49), (50, 48), (57, 53), (54, 55), (56, 57), 0.815x+9.711, 1.200, 2.304, 1.518
(48, 57), (53, 48), (54, 53), (48, 49), (54, 52), (54, 50), (51, 53), (52, 56), (57, 53), (49, 51), -0.087x+56.750, 3.400, 7.299, 2.702
(57, 53), (54, 53), (52, 49), (48, 54), (57, 56), (49, 48), (54, 53), (57, 55), (56, 57), (51, 55), 0.444x+29.551, 2.400, 5.390, 2.322
(49, 50), (50, 57), (50, 56), (53, 51), (57, 53), (51, 55), (49, 50), (55, 48), (49, 50), (56, 55), 0.017x+51.604, 3.400, 8.647, 2.941
(50, 57), (52, 56), (49, 48), (52, 55), (50, 50), (51, 49), (55, 53), (51, 50), (57, 50), (48, 57), -0.203x+62.955, 3.600, 10.776, 3.283
(50, 52), (51, 53), (53, 55), (55, 57), (57, 57), (50, 52), (51, 53), (53, 55), (55, 57), (57, 57), 0.768x+13.927, 1.600, 0.288, 0.536
(48, 49), (52, 52), (52, 53), (54, 55), (54, 54), (48, 49), (50, 48), (57, 53), (54, 55), (56, 57), 0.815x+9.711, 1.200, 2.304, 1.518
(48, 57), (53, 48), (54, 53), (48, 49), (54, 52), (54, 50), (51, 53), (52, 56), (57, 53), (49, 51), -0.087x+56.750, 3.400, 7.299, 2.702
(57, 53), (54, 53), (52, 49), (48, 54), (57, 56), (49, 48), (54, 53), (57, 55), (56, 57), (51, 55), 0.444x+29.551, 2.400, 5.390, 2.322
(49, 50), (50, 57), (50, 56), (53, 51), (57, 53), (51, 55), (49, 50), (55, 48), (49, 50), (56, 55), 0.017x+51.604, 3.400, 8.647, 2.941
(50, 57), (52, 56), (49, 48), (52, 55), (50, 50), (51, 49), (55, 53), (51, 50), (57, 50), (48, 57), -0.203x+62.955, 3.600, 10.776, 3.283
(50, 52), (51, 53), (53, 55), (55, 57), (57, 57), (50, 52), (51, 53), (53, 55), (55, 57), (57, 57), 0.768x+13.927, 1.600, 0.288, 0.536
```

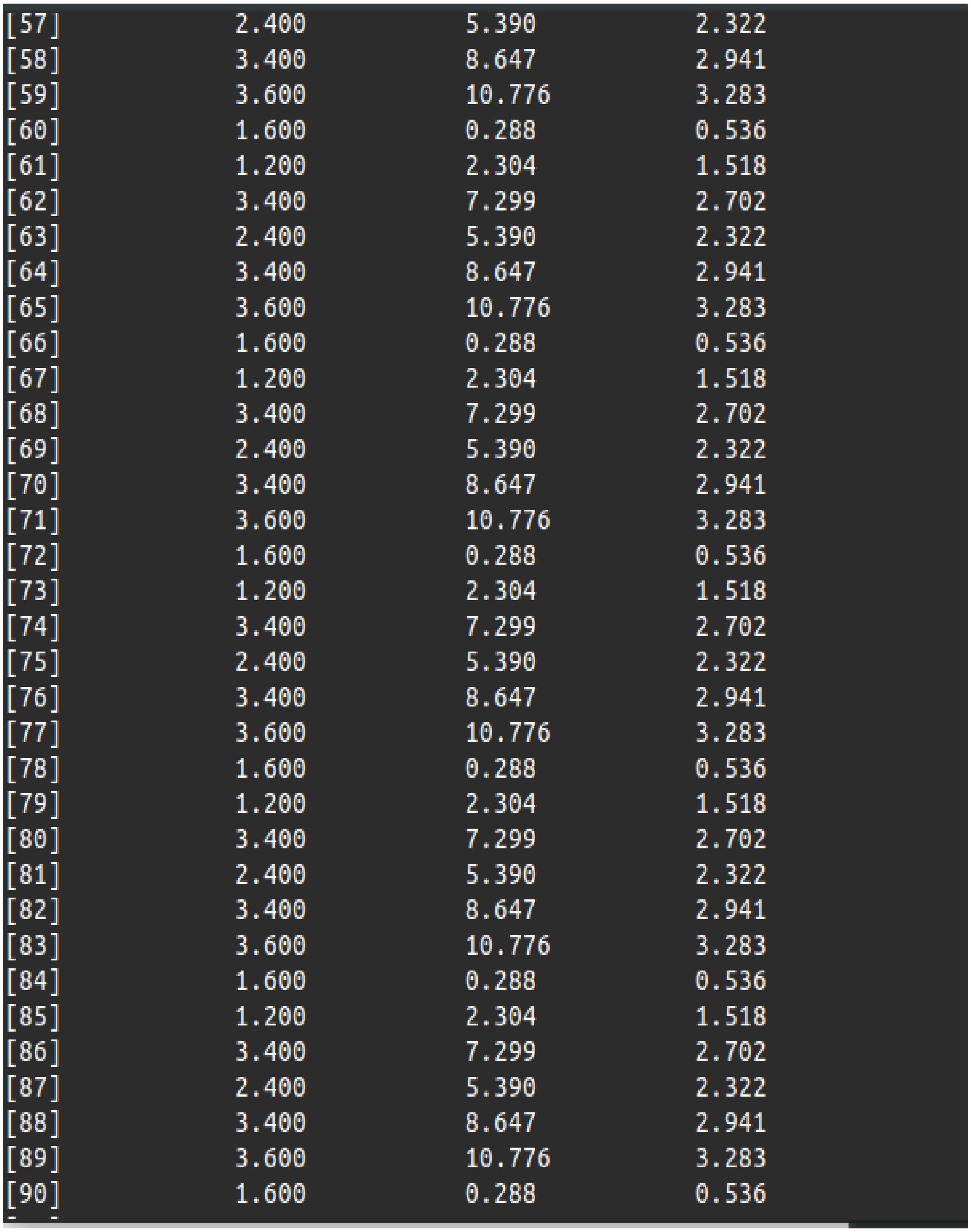
Console:



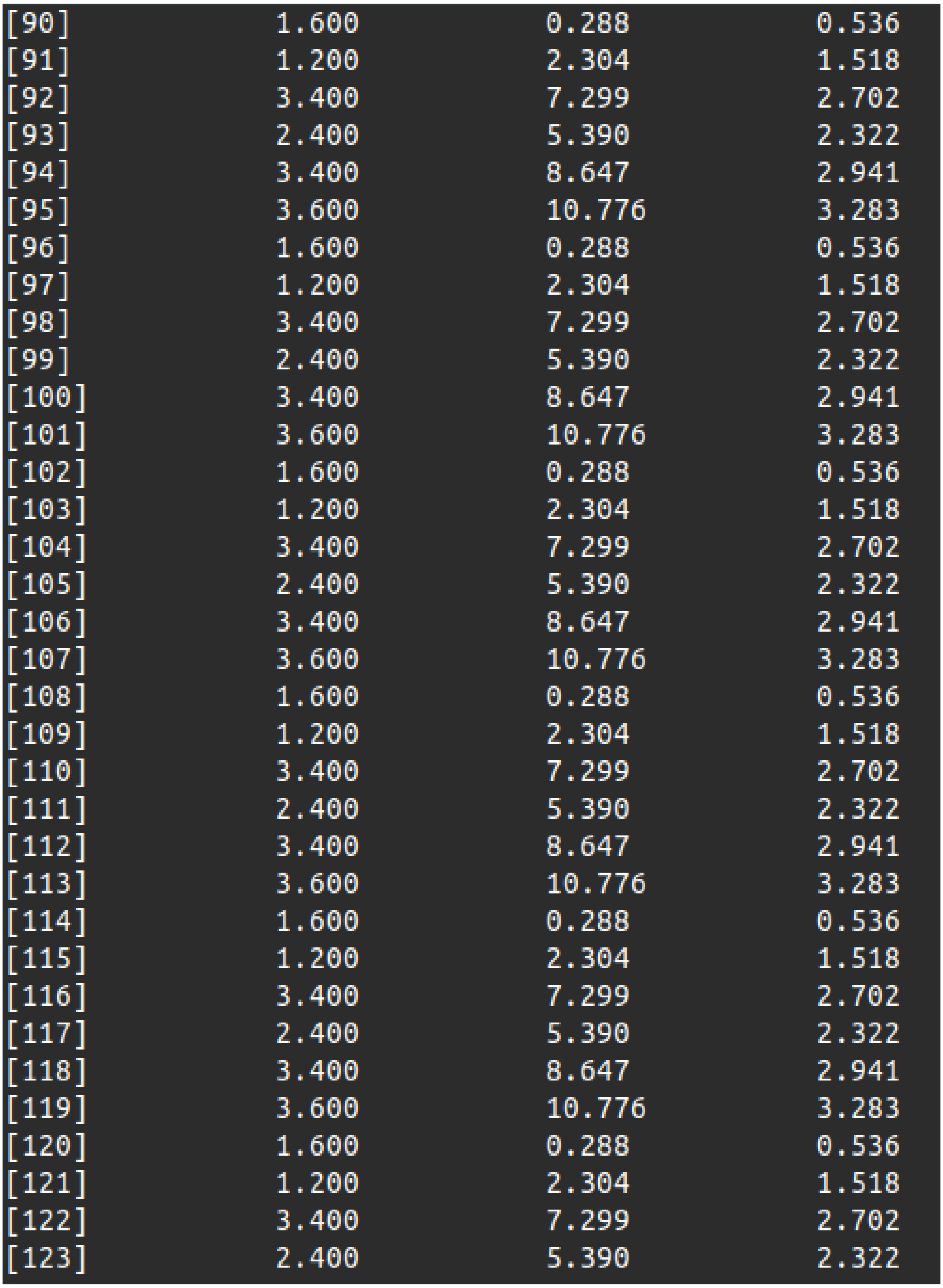
```
omer@omer:~/Desktop/HW02/HW02$ make
gcc -o program program.c -lm
./program -i inputPath.txt -o outputPath.txt
P1 Total read bytes : 3000
P1 Total read lines : 150
P1 Signals : SIGUSR1, SIGUSR2, SIGTERM

INDEX           MAE             MSE             RMSE
---------------------------------------------------------
[0]             1.600           0.288           0.536
[1]             1.200           2.304           1.518
[2]             3.400           7.299           2.702
[3]             2.400           5.390           2.322
[4]             3.400           8.647           2.941
[5]             3.600           10.776          3.283
[6]             1.600           0.288           0.536
[7]             1.200           2.304           1.518
[8]             3.400           7.299           2.702
[9]             2.400           5.390           2.322
[10]            3.400           8.647           2.941
[11]            3.600           10.776          3.283
[12]            1.600           0.288           0.536
[13]            1.200           2.304           1.518
[14]            3.400           7.299           2.702
[15]            2.400           5.390           2.322
[16]            3.400           8.647           2.941
[17]            3.600           10.776          3.283
[18]            1.600           0.288           0.536
[19]            1.200           2.304           1.518
[20]            3.400           7.299           2.702
[21]            2.400           5.390           2.322
[22]            3.400           8.647           2.941
[23]            3.600           10.776          3.283
[24]            1.600           0.288           0.536
```



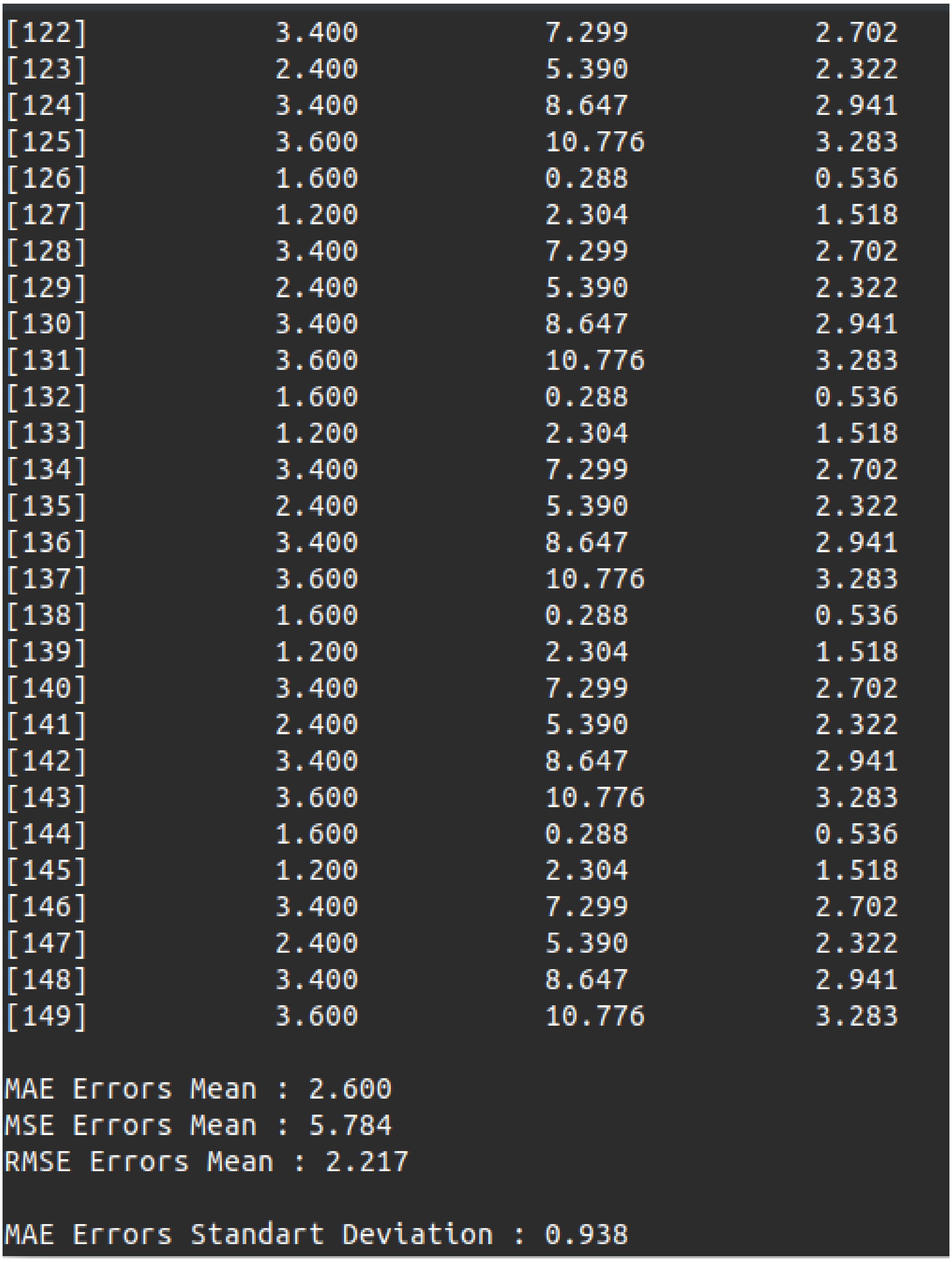
```
[24]            1.600           0.288           0.536
[25]            1.200           2.304           1.518
[26]            3.400           7.299           2.702
[27]            2.400           5.390           2.322
[28]            3.400           8.647           2.941
[29]            3.600           10.776          3.283
[30]            1.600           0.288           0.536
[31]            1.200           2.304           1.518
[32]            3.400           7.299           2.702
[33]            2.400           5.390           2.322
[34]            3.400           8.647           2.941
[35]            3.600           10.776          3.283
[36]            1.600           0.288           0.536
[37]            1.200           2.304           1.518
[38]            3.400           7.299           2.702
[39]            2.400           5.390           2.322
[40]            3.400           8.647           2.941
[41]            3.600           10.776          3.283
[42]            1.600           0.288           0.536
[43]            1.200           2.304           1.518
[44]            3.400           7.299           2.702
[45]            2.400           5.390           2.322
[46]            3.400           8.647           2.941
[47]            3.600           10.776          3.283
[48]            1.600           0.288           0.536
[49]            1.200           2.304           1.518
[50]            3.400           7.299           2.702
[51]            2.400           5.390           2.322
[52]            3.400           8.647           2.941
[53]            3.600           10.776          3.283
[54]            1.600           0.288           0.536
[55]            1.200           2.304           1.518
[56]            3.400           7.299           2.702
[57]            2.400           5.390           2.322
```



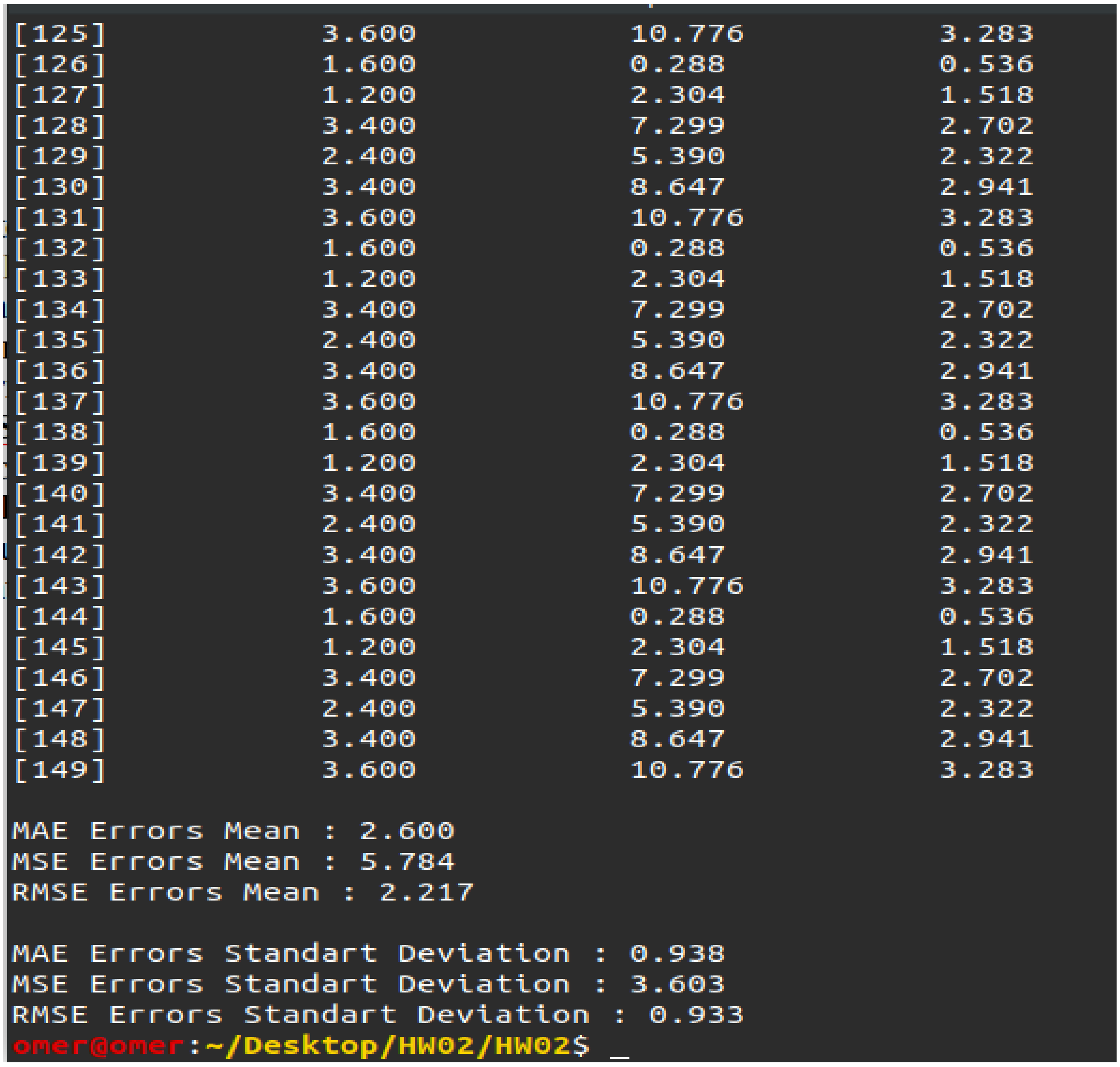
```
[57]            2.400           5.390           2.322
[58]            3.400           8.647           2.941
[59]            3.600           10.776          3.283
[60]            1.600           0.288           0.536
[61]            1.200           2.304           1.518
[62]            3.400           7.299           2.702
[63]            2.400           5.390           2.322
[64]            3.400           8.647           2.941
[65]            3.600           10.776          3.283
[66]            1.600           0.288           0.536
[67]            1.200           2.304           1.518
[68]            3.400           7.299           2.702
[69]            2.400           5.390           2.322
[70]            3.400           8.647           2.941
[71]            3.600           10.776          3.283
[72]            1.600           0.288           0.536
[73]            1.200           2.304           1.518
[74]            3.400           7.299           2.702
[75]            2.400           5.390           2.322
[76]            3.400           8.647           2.941
[77]            3.600           10.776          3.283
[78]            1.600           0.288           0.536
[79]            1.200           2.304           1.518
[80]            3.400           7.299           2.702
[81]            2.400           5.390           2.322
[82]            3.400           8.647           2.941
[83]            3.600           10.776          3.283
[84]            1.600           0.288           0.536
[85]            1.200           2.304           1.518
[86]            3.400           7.299           2.702
[87]            2.400           5.390           2.322
[88]            3.400           8.647           2.941
[89]            3.600           10.776          3.283
[90]            1.600           0.288           0.536
```



```
[90]              1.600             0.288             0.536
[91]              1.200             2.304             1.518
[92]              3.400             7.299             2.702
[93]              2.400             5.390             2.322
[94]              3.400             8.647             2.941
[95]              3.600             10.776            3.283
[96]              1.600             0.288             0.536
[97]              1.200             2.304             1.518
[98]              3.400             7.299             2.702
[99]              2.400             5.390             2.322
[100]             3.400             8.647             2.941
[101]             3.600             10.776            3.283
[102]             1.600             0.288             0.536
[103]             1.200             2.304             1.518
[104]             3.400             7.299             2.702
[105]             2.400             5.390             2.322
[106]             3.400             8.647             2.941
[107]             3.600             10.776            3.283
[108]             1.600             0.288             0.536
[109]             1.200             2.304             1.518
[110]             3.400             7.299             2.702
[111]             2.400             5.390             2.322
[112]             3.400             8.647             2.941
[113]             3.600             10.776            3.283
[114]             1.600             0.288             0.536
[115]             1.200             2.304             1.518
[116]             3.400             7.299             2.702
[117]             2.400             5.390             2.322
[118]             3.400             8.647             2.941
[119]             3.600             10.776            3.283
[120]             1.600             0.288             0.536
[121]             1.200             2.304             1.518
[122]             3.400             7.299             2.702
[123]             2.400             5.390             2.322
```



```
[122]            3.400            7.299            2.702
[123]            2.400            5.390            2.322
[124]            3.400            8.647            2.941
[125]            3.600            10.776           3.283
[126]            1.600            0.288            0.536
[127]            1.200            2.304            1.518
[128]            3.400            7.299            2.702
[129]            2.400            5.390            2.322
[130]            3.400            8.647            2.941
[131]            3.600            10.776           3.283
[132]            1.600            0.288            0.536
[133]            1.200            2.304            1.518
[134]            3.400            7.299            2.702
[135]            2.400            5.390            2.322
[136]            3.400            8.647            2.941
[137]            3.600            10.776           3.283
[138]            1.600            0.288            0.536
[139]            1.200            2.304            1.518
[140]            3.400            7.299            2.702
[141]            2.400            5.390            2.322
[142]            3.400            8.647            2.941
[143]            3.600            10.776           3.283
[144]            1.600            0.288            0.536
[145]            1.200            2.304            1.518
[146]            3.400            7.299            2.702
[147]            2.400            5.390            2.322
[148]            3.400            8.647            2.941
[149]            3.600            10.776           3.283

MAE Errors Mean : 2.600
MSE Errors Mean : 5.784
RMSE Errors Mean : 2.217

MAE Errors Standart Deviation : 0.938
```



```
[125]               3.600               10.776              3.283
[126]               1.600               0.288               0.536
[127]               1.200               2.304               1.518
[128]               3.400               7.299               2.702
[129]               2.400               5.390               2.322
[130]               3.400               8.647               2.941
[131]               3.600               10.776              3.283
[132]               1.600               0.288               0.536
[133]               1.200               2.304               1.518
[134]               3.400               7.299               2.702
[135]               2.400               5.390               2.322
[136]               3.400               8.647               2.941
[137]               3.600               10.776              3.283
[138]               1.600               0.288               0.536
[139]               1.200               2.304               1.518
[140]               3.400               7.299               2.702
[141]               2.400               5.390               2.322
[142]               3.400               8.647               2.941
[143]               3.600               10.776              3.283
[144]               1.600               0.288               0.536
[145]               1.200               2.304               1.518
[146]               3.400               7.299               2.702
[147]               2.400               5.390               2.322
[148]               3.400               8.647               2.941
[149]               3.600               10.776              3.283

MAE Errors Mean : 2.600
MSE Errors Mean : 5.784
RMSE Errors Mean : 2.217

MAE Errors Standart Deviation : 0.938
MSE Errors Standart Deviation : 3.603
RMSE Errors Standart Deviation : 0.933
omer@omer:~/Desktop/HW02/HW02$ _
```

Notes:

- To run the program, I write a Makefile which exactly compiles and runs **program.c** using “***make***” command. Input file name is “***inputPath.txt***” and output file name is “***outputPath.txt***”.
- I send my input files and Makefile in .zip file.
- No errors, no warnings I caught showed in test results.
- All lockings works using signals.
- In my input file there are 150 lines to test which includes 20 character in each line (ignored new line characters).
- (*) If there is something wrong with deadlock, please kill the processes and restart to compile and run.