Hochschule Karlsruhe
University of
Applied Sciences
HKA

Faculty of Electrical Engineering and Information Technology (EIT)

Sensor Systems Technology

ST3M71 Advanced Programming

WarmUp Part B

Prof. Dr. Th. Leize

Yunus Emin Yazici (68571)

A C++ program is written which able to, **read the text lines from** file, **print the text lines to:** screen, file **sort the lines alphabetically.**

1. First version of code done without using the string class and ready done containers.

2. Second version of code done with using the string class STL/class library container (vector, list, queue, ...) <algorithm>

Comments are added to source codes.

Explanation of the files:

1. testfile.txt ------ 1000 lines of data

2. firsttask.cpp ---- first version of code

3. sortedfile.txt ------ sorted text output

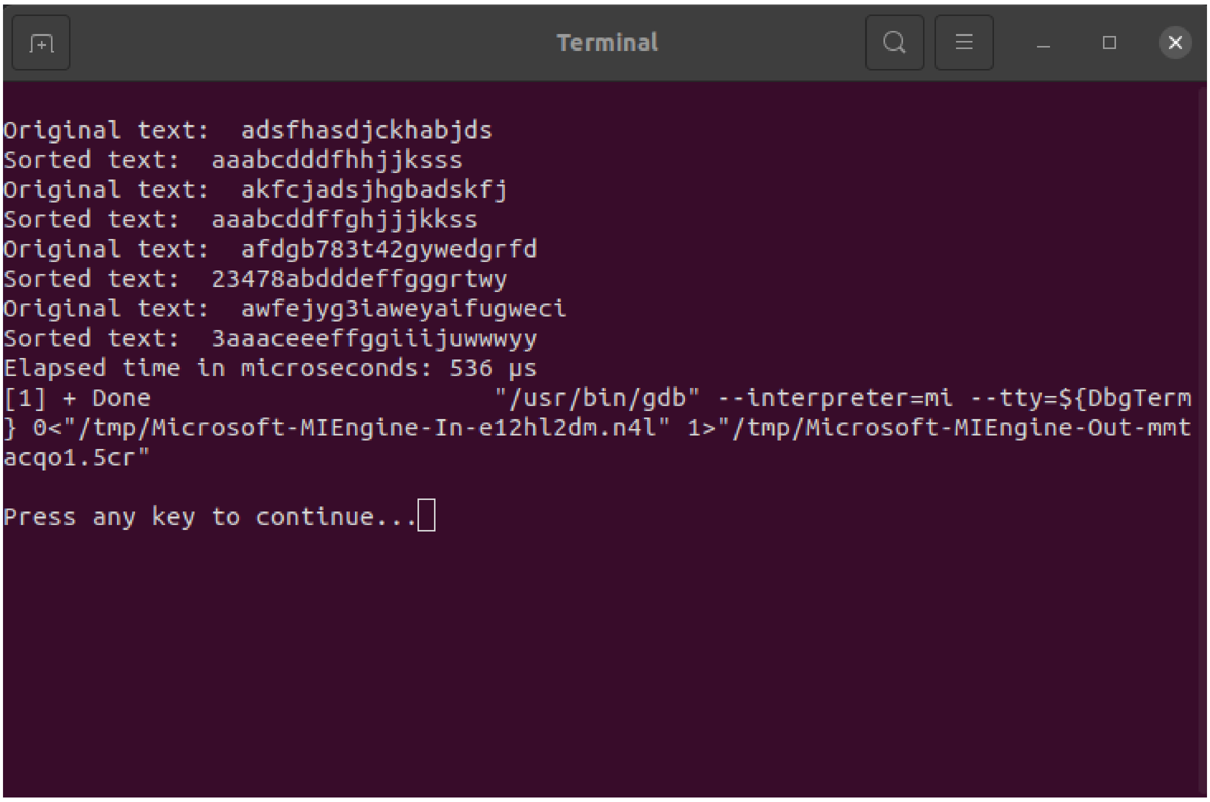
4. secondtask.cpp ------- second version of code

5. sortedfile2.txt ------ sorted text output

Screenshots of the output terminals

Note: To illustrate the code works successfully small data set is used here. For the time measurement, 1000 lines data is used. Please see page 5 and 6.

First version:

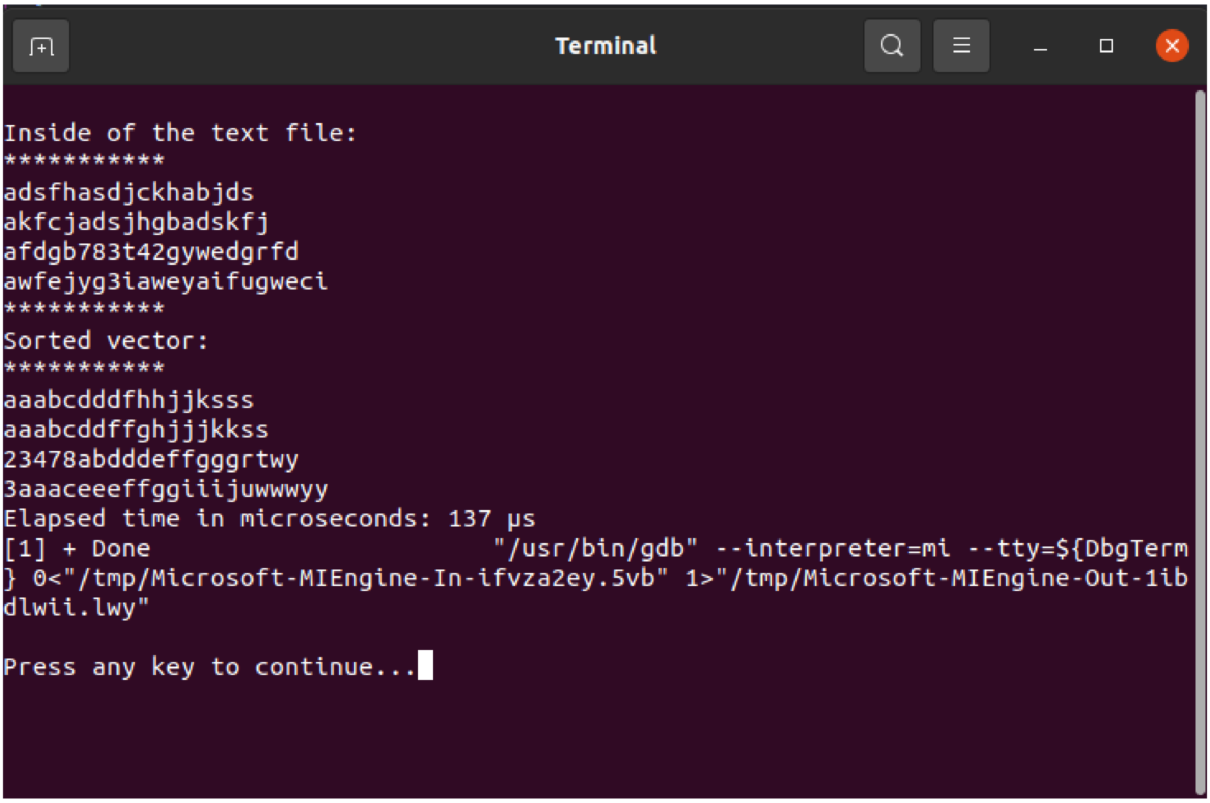


```
Original text:  adsfhasdjckhabjds
Sorted text:  aaabcdddfhhjjksss
Original text:  akfcjadsjhgbadskfj
Sorted text:  aaabcddffghjjjkkss
Original text:  afdgb783t42gywedgrfd
Sorted text:  23478abdddeffgggrtwy
Original text:  awfejyg3iaweyaifugweci
Sorted text:  3aaaceeeffggiiijuwwwyy
Elapsed time in microseconds: 536 µs
[1] + Done                       "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm
} 0<"/tmp/Microsoft-MIEngine-In-e12hl2dm.n4l" 1>"/tmp/Microsoft-MIEngine-Out-mmt
acqo1.5cr"

Press any key to continue...
```

Original and sorted text printed to the screen.

Second Version:



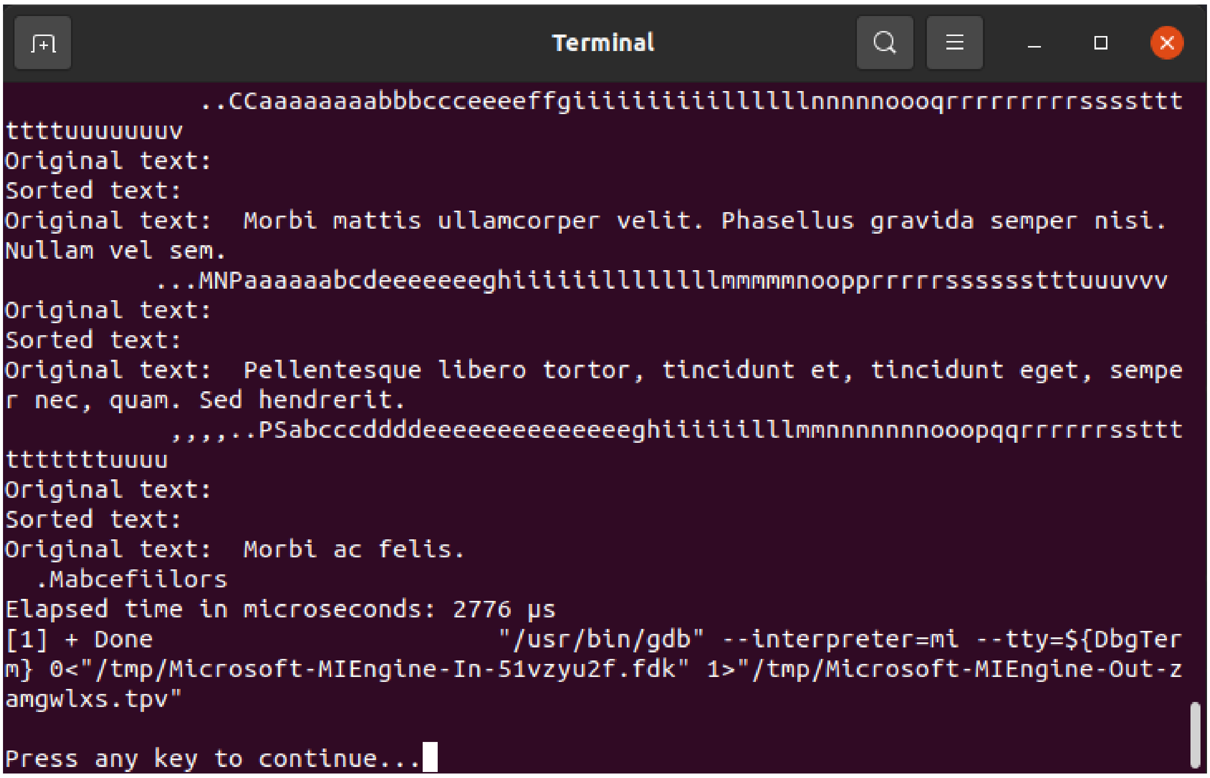
```
Inside of the text file:
***********
adsfhasdjckhabjds
akfcjadsjhgbadskfj
afdgb783t42gywedgrfd
awfejyg3iaweyaifugweci
***********
Sorted vector:
***********
aaabcdddfhhjjksss
aaabcddffghjjjkkss
23478abdddeffgggrtwy
3aaaceeeffggiiijuwwwyy
Elapsed time in microseconds: 137 µs
[1] + Done                       "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm
} 0<"/tmp/Microsoft-MIEngine-In-ifvza2ey.5vb" 1>"/tmp/Microsoft-MIEngine-Out-1ib
dlwii.lwy"

Press any key to continue...
```

Original and sorted text printed to the screen.

Screenshots for time measurement:

First version:

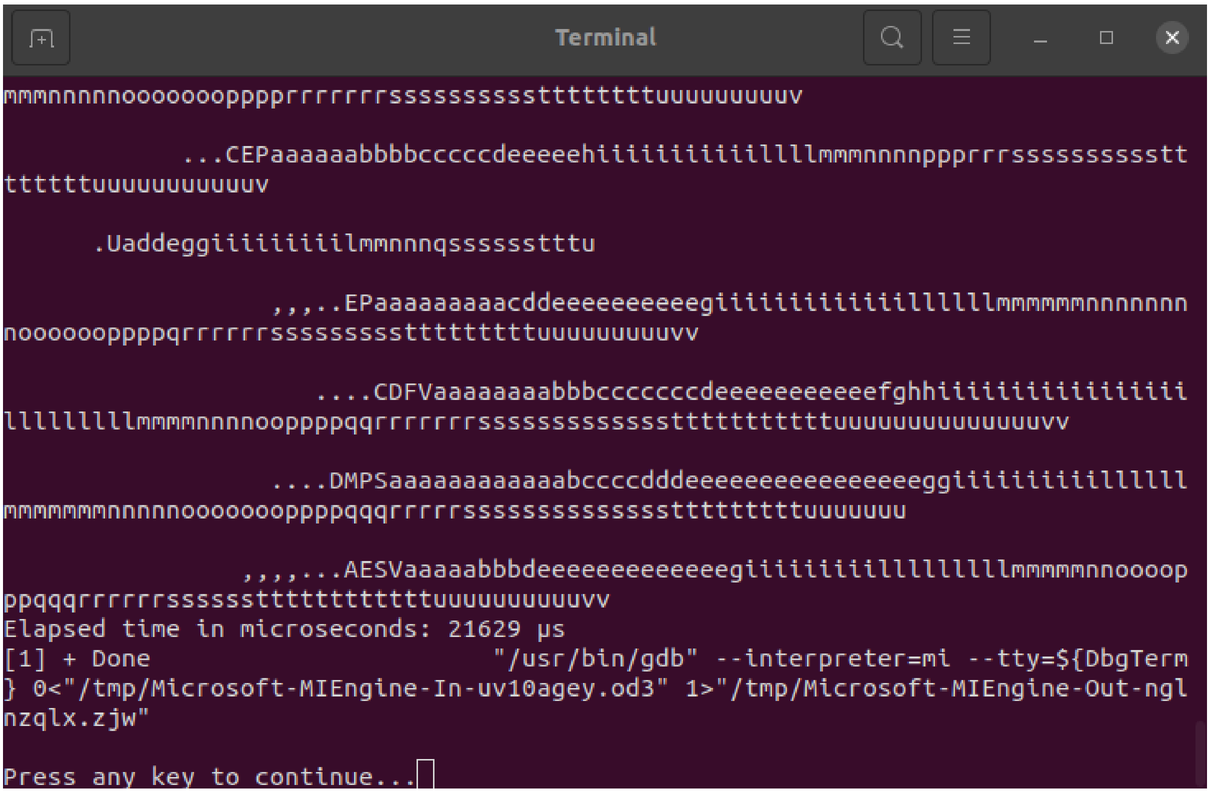


```
             ..CCaaaaaaaabbbccceeeeffgiiiiiiiiiilllllllnnnnnoooqrrrrrrrrrsssstttt
ttttuuuuuuuv
Original text:
Sorted text:
Original text:  Morbi mattis ullamcorper velit. Phasellus gravida semper nisi.
Nullam vel sem.
          ...MNPaaaaaabcdeeeeeeeeghiiiiiiilllllllmmmmmnoopprrrrrsssssstttuuuvvv
Original text:
Sorted text:
Original text:  Pellentesque libero tortor, tincidunt et, tincidunt eget, sempe
r nec, quam. Sed hendrerit.
           ,,,,..PSabcccddddeeeeeeeeeeeeeeeghiiiiiiilllmmnnnnnnnnoooopqqrrrrrrssttt
tttttttuuuu
Original text:
Sorted text:
Original text:  Morbi ac felis.
  .Mabcefiilors
Elapsed time in microseconds: 2776 µs
[1] + Done                       "/usr/bin/gdb" --interpreter=mi --tty=${DbgTer
m} 0<"/tmp/Microsoft-MIEngine-In-51vzyu2f.fdk" 1>"/tmp/Microsoft-MIEngine-Out-z
amgwlxs.tpv"

Press any key to continue...
```

Time measurement for the first version of code is 2776us.

Second version:



```
mmmnnnnnoooooooppppprrrrrrssssssssssttttttttuuuuuuuuuv

            ...CEPaaaaaabbbbcccccdeeeeehiiiiiiiiiiiiillllmmmnnnnnppprrrsssssssssstt
tttttttuuuuuuuuuuuuv

      .Uaddeggiiiiiiiiilmmnnnqsssssstttu

                  ,,,,..EPaaaaaaaaacddeeeeeeeeeeegiiiiiiiiiiiiiilllllmmmmmmnnnnnnn
nooooooppppqrrrrrrssssssssstttttttttuuuuuuuuuuvv

                     ....CDFVaaaaaaaabbbccccccdeeeeeeeeeeeefghhiiiiiiiiiiiiiiiii
lllllllllmmmmnnnnnooppppqqrrrrrrsssssssssssssttttttttttttuuuuuuuuuuuuuuuvv

                  ....DMPSaaaaaaaaaaaabccccdddeeeeeeeeeeeeeeeeeeggiiiiiiiiiiillllll
mmmmmmmnnnnnoooooooppppqqqrrrrrsssssssssssssstttttttttuuuuuuu

                ,,,,,...AESVaaaaabbbdeeeeeeeeeeeeeeegiiiiiiiiiillllllllmmmmmnnoooop
ppqqqrrrrrrssssssttttttttttttuuuuuuuuuuuvv
Elapsed time in microseconds: 21629 µs
[1] + Done                       "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm
} 0<"/tmp/Microsoft-MIEngine-In-uv10agey.od3" 1>"/tmp/Microsoft-MIEngine-Out-ngl
nzqlx.zjw"

Press any key to continue...
```

Time measurement for the first version of code is 21629us.

Conclusion:

As we seen in the screenshots first version of code much more faster than second version of code. Without using the string class and containers the program works much more faster than second version.