Gebze Technical University

Computer Engineering Department

CSE 344 – System Programming

Homework-4 Report

Muhammet Akkurt

1901042644

Purpose and Details of the Assignment:

This assignment is a multi-threaded program designed to copy files from one directory to another. The program is built using POSIX threads to handle multiple file operations simultaneously.

The system architecture consists of a manager thread that populates a shared buffer with file metadata and multiple worker threads that perform copying of files based on data retrieved from the buffer. Synchronization between these threads is achieved using mutexes and condition variables to manage access to the shared buffer and coordinate the producer-consumer relationship.

The program handles different types of files, including regular files and FIFOs. Regular files are directly copied, whereas FIFOs are recreated at the destination.

Directories are processed recursively, ensuring that all nested files and subdirectories are copied. The program replicates directory structures accurately at the destination.

Mutexes are used to ensure exclusive access to the shared buffer by one thread at a time, while condition variables manage thread synchronization, signaling threads to sleep or wake up based on the buffer's state of being empty or full.

Pseudocode:

```
Main Program:

   Initialize resources (buffer, threads)

   Start timing

   Create manager thread with source and destination directories

   Create worker threads

   Wait for all threads to complete

   Stop timing

   Print performance statistics

   Clean up resources

Manager Thread:

   For each file in the source directory:

      Wait until there is space in the buffer

      Lock the buffer

      Add file to buffer

      Signal worker threads that buffer is not empty

      Unlock the buffer

   Indicate completion and exit

Worker Thread:

   While not done:

      Lock the buffer

      Wait for buffer to be not empty

      Retrieve file information from buffer

      If file is a FIFO:

         Create FIFO at destination

      Else:

         Copy file from source to destination

      Update copied bytes and file count

      Unlock the buffer

   Exit thread

Signal Handling:

   On SIGINT or SIGTERM:

      Set termination flag

      Signal all conditions to ensure no thread is left waiting
```
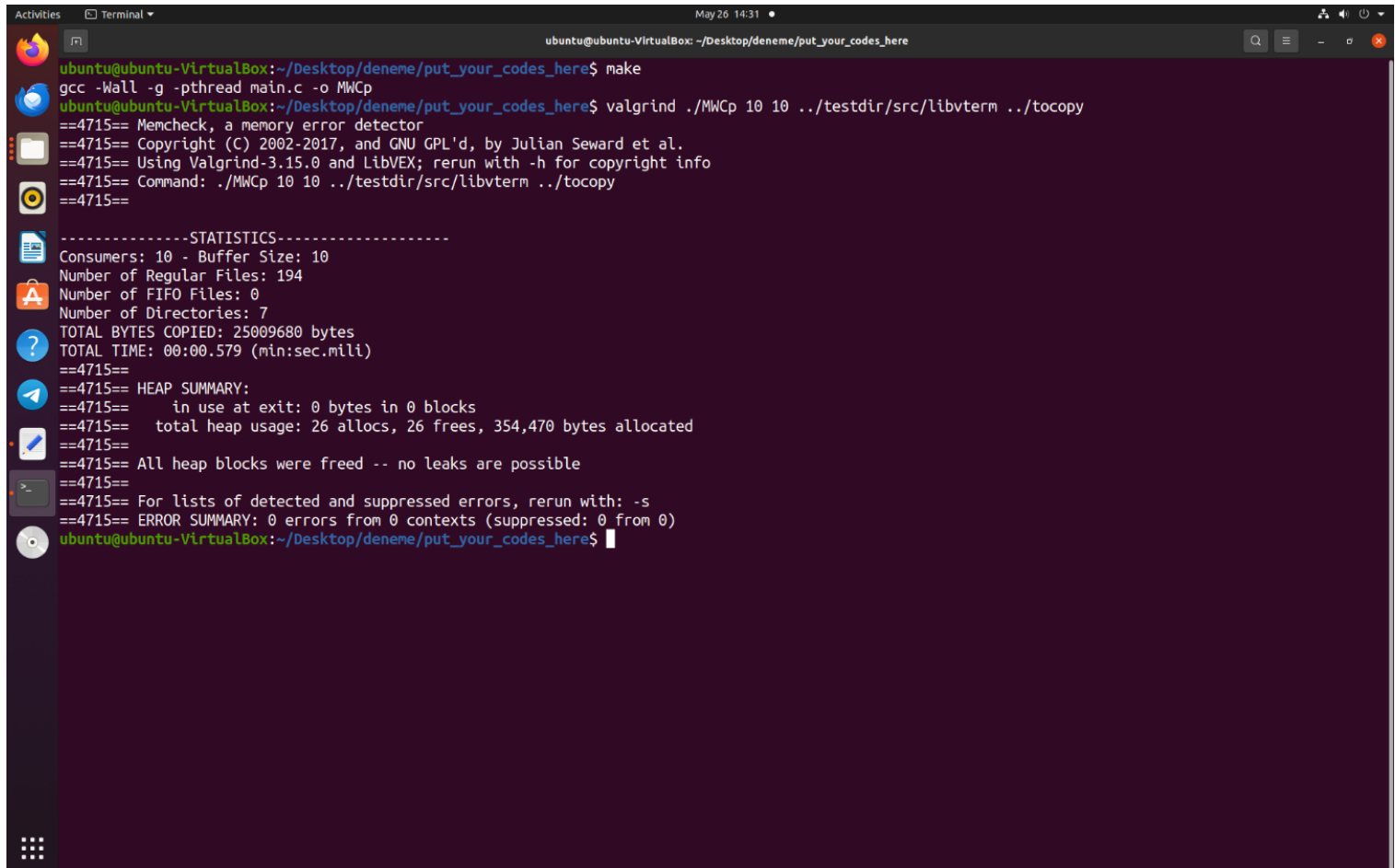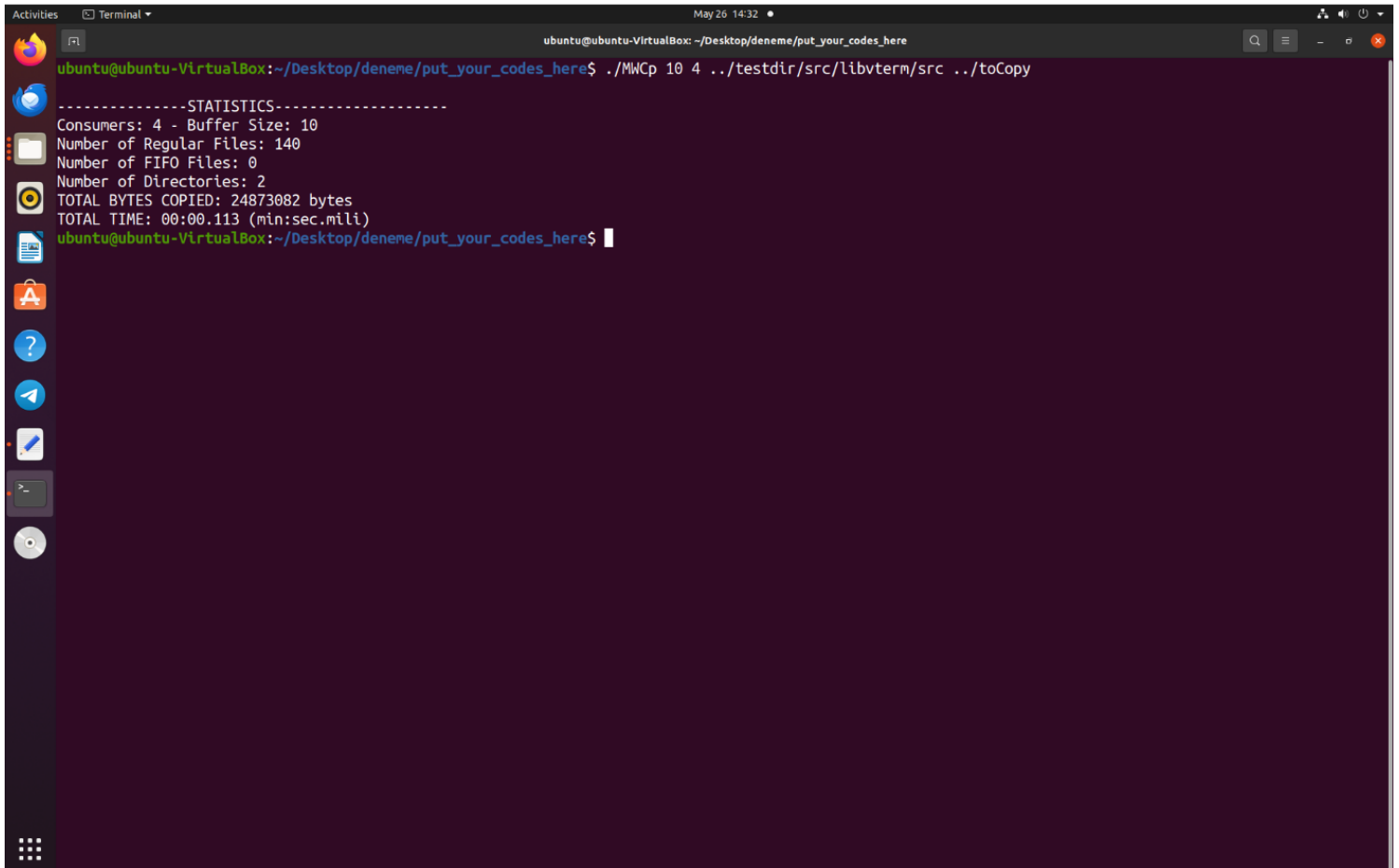
Test Scenarios:

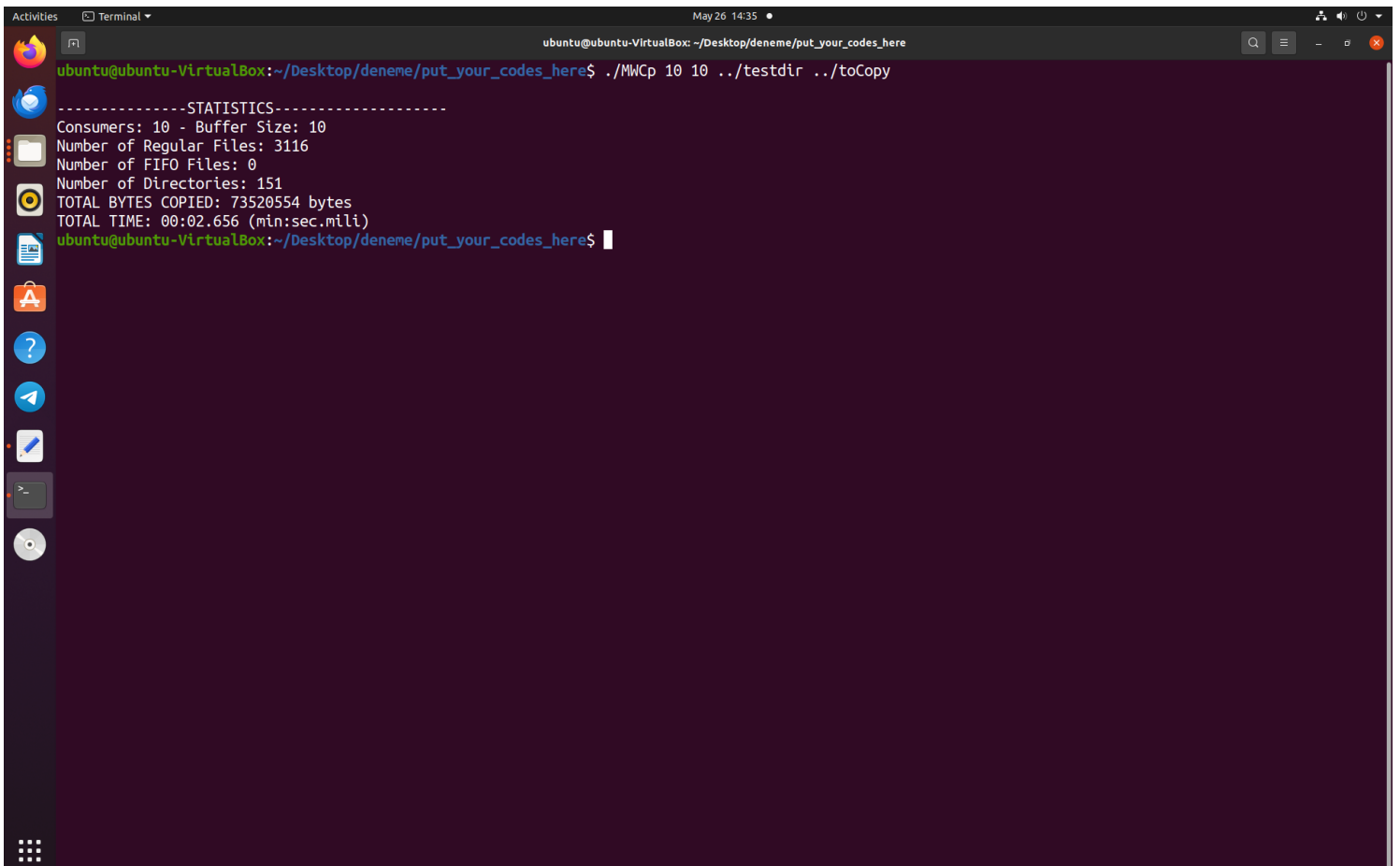- Test1: valgrind ./MWCp 10 10 ../testdir/src/libvterm ../tocopy

- Test2: ./MWCp 10 4 ../testdir/src/libvterm/ src ../toCopy


```
ubuntu@ubuntu-VirtualBox:~/Desktop/deneme/put_your_codes_here$ ./MWCp 10 4 ../testdir/src/libvterm/src ../toCopy

---------------STATISTICS-------------------
Consumers: 4 - Buffer Size: 10
Number of Regular Files: 140
Number of FIFO Files: 0
Number of Directories: 2
TOTAL BYTES COPIED: 24873082 bytes
TOTAL TIME: 00:00.113 (min:sec.mili)
ubuntu@ubuntu-VirtualBox:~/Desktop/deneme/put_your_codes_here$
```

- Test3: ./MWCp 10 10 ../testdir ../toCopy


```
ubuntu@ubuntu-VirtualBox:~/Desktop/deneme/put_your_codes_here$ ./MWCp 10 10 ../testdir ../toCopy

---------------STATISTICS-------------------
Consumers: 10 - Buffer Size: 10
Number of Regular Files: 3116
Number of FIFO Files: 0
Number of Directories: 151
TOTAL BYTES COPIED: 73520554 bytes
TOTAL TIME: 00:02.656 (min:sec.mili)
ubuntu@ubuntu-VirtualBox:~/Desktop/deneme/put_your_codes_here$
```

- Test4: valgrind ./MWCp 10 10 ../testdir ../tocopy
  (Signal Handling CTRL+C)

```
ubuntu@ubuntu-VirtualBox:~/Desktop/deneme/put_your_codes_here$ valgrind ./MWCp 10 10 ../testdir ../tocopy
==4876== Memcheck, a memory error detector
==4876== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==4876== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==4876== Command: ./MWCp 10 10 ../testdir ../tocopy
==4876==
^C
---------------STATISTICS--------------------
Consumers: 10 - Buffer Size: 10
Number of Regular Files: 597
Number of FIFO Files: 0
Number of Directories: 38
TOTAL BYTES COPIED: 4308941 bytes
TOTAL TIME: 00:01.087 (min:sec.mili)
==4876==
==4876== HEAP SUMMARY:
==4876==     in use at exit: 0 bytes in 0 blocks
==4876==   total heap usage: 57 allocs, 57 frees, 1,371,766 bytes allocated
==4876==
==4876== All heap blocks were freed -- no leaks are possible
==4876==
==4876== For lists of detected and suppressed errors, rerun with: -s
==4876== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
ubuntu@ubuntu-VirtualBox:~/Desktop/deneme/put_your_codes_here$
```

- add and test two FIFO files in the "testdir" folder
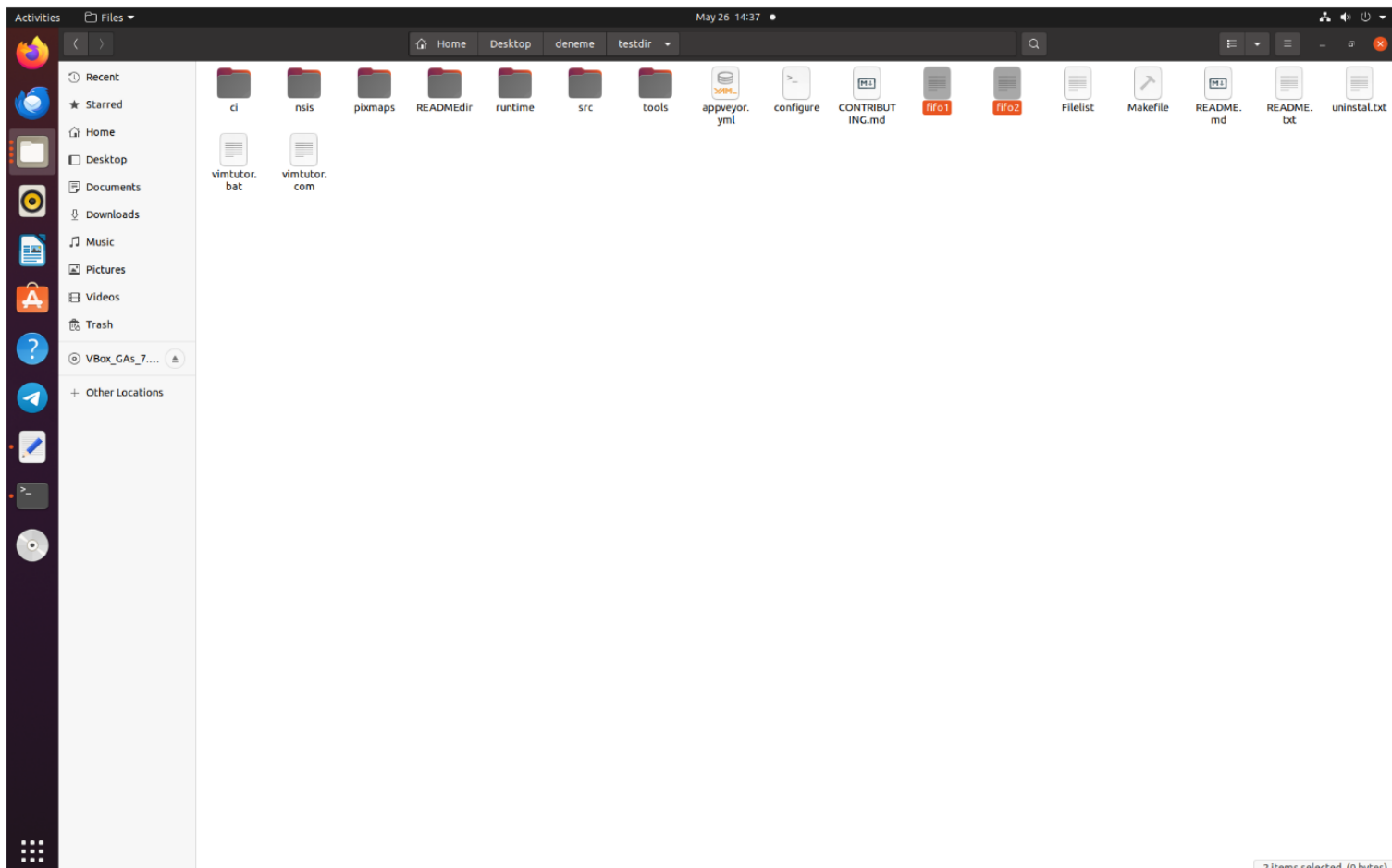
Test5: valgrind ./MWCp 4 4 ../testdir ../tocopy

```
ubuntu@ubuntu-VirtualBox:~/Desktop/deneme/put_your_codes_here$ valgrind ./MWCp 4 4 ../testdir ../tocopy
==4944== Memcheck, a memory error detector
==4944== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==4944== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==4944== Command: ./MWCp 4 4 ../testdir ../tocopy
==4944==

--------------STATISTICS--------------------
Consumers: 4 - Buffer Size: 4
Number of Regular Files: 3116
Number of FIFO Files: 2
Number of Directories: 151
TOTAL BYTES COPIED: 73520554 bytes
TOTAL TIME: 00:03.256 (min:sec.mili)
==4944==
==4944== HEAP SUMMARY:
==4944==     in use at exit: 0 bytes in 0 blocks
==4944==   total heap usage: 164 allocs, 164 frees, 5,025,998 bytes allocated
==4944==
==4944== All heap blocks were freed -- no leaks are possible
==4944==
==4944== For lists of detected and suppressed errors, rerun with: -s
==4944== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
ubuntu@ubuntu-VirtualBox:~/Desktop/deneme/put_your_codes_here$
```