

# **CSE 344**

# **SYSTEM PROGRAMMING**

## **HW 4 Report**

Emre Güven

1901042621

## INTRODUCTION

This project is a multi-threaded file copying utility designed to traverse a source directory and replicate its structure and contents to a destination directory using a buffer and worker threads for efficient file handling. It maintains statistics on the types of files copied and the total bytes transferred.

To compile program in shell:     \$> make

To run program correctly:       \$> ./MWCp <task\_buffer\_size> <#of workers> <source\_folder>  
                                  <destination\_folder>

To clean unnecessary files:     \$> make clean

## USED MUTEXES AND CONDITION VARIABLES

### Mutexes

**buffer\_mutex:** Protects access to the task buffer and related variables (buffer\_front, buffer\_rear, buffer\_count).

**stats\_mutex:** Protects access to the statistics variables (num\_regular\_files, num\_fifo\_files, num\_directories, num\_other\_files, total\_bytes\_copied).

**stdout\_mutex:** Synchronizes access to standard output.

### Condition Variables

**buffer\_not\_empty:** Signals worker threads that the buffer has at least one task available.

**buffer\_not\_full:** Signals the manager thread that there is space available in the buffer.

## USED TYPES AND GLOBAL VARIABLES

### Types

**file\_task\_t:** A structure representing a file copying task. It holds file descriptors for the source file and destination file.

**directory\_paths\_t:** A structure representing the source and destination directories.

### Global Variables

**task\_buffer:** A dynamically allocated buffer that holds the file copying tasks to be processed by worker threads.

**number\_of\_workers:** Stores the number of worker threads that will process tasks from the task buffer.

**buffer\_size:** Defines the size of the task buffer, determining how many tasks can be queued at any given time.

**buffer\_front, buffer\_rear, buffer\_count:** Indexes and count for handling buffer.

**done\_flag:** A flag indicating whether the copying process should be terminated, used for handling signals safely.

**num\_regular\_files, num\_fifo\_files, num\_directories, num\_other\_files:** Counters for different types of files and directories processed.

**total\_bytes\_copied:** Accumulates the total number of bytes copied across all files.

## PROGRAM STRUCTURE AS PSEUDOCODE

The program is a multi-threaded file copying utility that initializes global variables and sets up necessary signal handlers. The **main function** parses command-line arguments, allocates memory for the task buffer, creates manager and worker threads, and measures the execution time. The **manager\_thread** traverses the source directory, adds tasks to the buffer, and signals completion. **Worker threads** process these tasks by copying files and updating statistics. Each function is explained as pseudocode.

### Main function

- Parse command line arguments and check them
- Allocate memory for task\_buffer and initialize it
- Call setup\_signal\_handler
- Create manager thread
- Create worker threads
- Measure time
- Join manager and worker threads
- Print statistics
- Free resources and destroy mutexes and condition variables

### worker\_thread function

- Loop indefinitely
  - Lock buffer\_mutex
  - Wait for buffer\_not\_empty if buffer\_count is 0 and done\_flag is not set
  - Break if buffer\_count is 0 and done\_flag is set
  - Get task from task\_buffer
  - Update buffer\_front and buffer\_count
  - Signal buffer\_not\_full
  - Unlock buffer\_mutex
  - Call copy\_file with current task

### copy\_file function

- Read from source\_fd and write to destination\_fd in chunks
- Handle read/write errors
- Close source\_fd and destination\_fd
- Lock stats\_mutex
- Update total\_bytes\_copied
- Unlock stats\_mutex

### manager\_thread function

- Call traverse\_directory with source\_folder and destination\_folder
- Lock buffer\_mutex
- Set done\_flag
- Broadcast buffer\_not\_empty and buffer\_not\_full
- Unlock buffer\_mutex

**traverse\_directory function**

- Open source\_folder directory
- Create destination\_folder if it does not exist
- For each entry in source\_folder
  - If entry is a directory and not "." or ".."
    - Call handle\_directory with source\_folder, destination\_folder, entry
  - Else Call handle\_file with source\_folder, destination\_folder, entry
- Close directory

**handle\_directory function**

- Construct new\_source and new\_dest paths
- Call traverse\_directory with new\_source and new\_dest to handle sub directory
- Lock stats\_mutex
- Increment num\_directories safely by using stats\_mutex
- Unlock stats\_mutex

**handle\_file function**

- Construct source\_file and destination\_file paths
- Open source\_file for reading
- Open destination\_file for writing
- Create file\_task\_t with file descriptors and file paths
- Lock buffer\_mutex
- Wait for buffer\_not\_full if buffer\_count is buffer\_size and done\_flag is not set
- Break if done\_flag is set
- Add task to task\_buffer
- Update buffer\_rear and buffer\_count
- Signal buffer\_not\_empty
- Unlock buffer\_mutex
- Call update\_statistics with entry

**update\_statistics function**

- Lock stats\_mutex
- Update file type counters based on entry type
- Unlock stats\_mutex

**setup\_signal\_handler function**

- Set signal handlers for SIGINT, SIGTERM, SIGQUIT, SIGHUP, SIGUSR1, SIGUSR2

**handle\_signal function**

- Set done\_flag
- Broadcast buffer\_not\_empty and buffer\_not\_full

**destroy\_mutexes\_and\_cond\_vars function**

- Destroy buffer\_mutex, stats\_mutex, stdout\_mutex
- Destroy buffer\_not\_empty and buffer\_not\_full

**close\_fd function**

- Close file descriptor and handle errors

## TESTS

**NOTE:** Printing the completion status of each file is not shown in some test screenshots because it may takes space in report. I have given shown examples below for one short number of folder entries and one CTRL+C example.

### Copying different folders completely (No printing)

```
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ make
gcc -Wall -std=gnu99 main.c -o MMCp -lpthread -lrt
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ valgrind --leak-check=full ./MMCp 10 10 ../testdir/src/libvterm ../tocopy
==9753== Memcheck, a memory error detector
==9753== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==9753== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==9753== Command: ./MMCp 10 10 ../testdir/src/libvterm ../tocopy
==9753==
-----STATISTICS-----
Consumers: 10 - Buffer Size: 10
Number of Regular Files: 194
Number of FIFO Files: 0
Number of Directories: 7
Number of Other Files: 0
TOTAL BYTES COPIED: 25009680
TOTAL TIME: 00:00.679 (min:sec.millis)
==9753==
==9753== HEAP SUMMARY:
==9753==      in use at exit: 0 bytes in 0 blocks
==9753==    total heap usage: 22 allocs, 22 frees, 287,184 bytes allocated
==9753==
==9753== All heap blocks were freed -- no leaks are possible
==9753==
==9753== For lists of detected and suppressed errors, rerun with: -s
==9753== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$
```

```
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ ./MMCp 10 10 ../testdir ../toCopy
-----STATISTICS-----
Consumers: 10 - Buffer Size: 10
Number of Regular Files: 3116
Number of FIFO Files: 0
Number of Directories: 151
Number of Other Files: 0
TOTAL BYTES COPIED: 73520554
TOTAL TIME: 00:00.233 (min:sec.millis)
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$
```

```
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ ./MMCp 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
Number of Other Files: 0
TOTAL BYTES COPIED: 24873082
TOTAL TIME: 00:00.087 (min:sec.millis)
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$
```

```
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ valgrind ./MMCp 12 5 ../testdir/src/libvterm ../toCopyA
==10282== Memcheck, a memory error detector
==10282== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==10282== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==10282== Command: ./MMCp 12 5 ../testdir/src/libvterm ../toCopyA
==10282==
-----STATISTICS-----
Consumers: 5 - Buffer Size: 12
Number of Regular Files: 194
Number of FIFO Files: 0
Number of Directories: 7
Number of Other Files: 0
TOTAL BYTES COPIED: 25009680
TOTAL TIME: 00:00.709 (min:sec.millis)
==10282==
==10282== HEAP SUMMARY:
==10282==      in use at exit: 0 bytes in 0 blocks
==10282==    total heap usage: 17 allocs, 17 frees, 289,896 bytes allocated
==10282==
==10282== All heap blocks were freed -- no leaks are possible
==10282==
==10282== For lists of detected and suppressed errors, rerun with: -s
==10282== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$
```

```
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ valgrind ./MMCp 7 10 ../testdir/runtime ../toCopyB
==11549== Memcheck, a memory error detector
==11549== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==11549== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==11549== Command: ./MMCp 7 10 ../testdir/runtime ../toCopyB
==11549==

-----STATISTICS-----
Consumers: 10 - Buffer Size: 7
Number of Regular Files: 1899
Number of FIFO Files: 0
Number of Directories: 98
Number of Other Files: 0
TOTAL BYTES COPIED: 24826550
TOTAL TIME: 00:01.1141 (min:sec.millis)
==11549==
==11549== HEAP SUMMARY:
==11549==   in use at exit: 0 bytes in 0 blocks
==11549==   total heap usage: 113 allocs, 113 frees, 3,252,936 bytes allocated
==11549==
==11549== All heap blocks were freed -- no leaks are possible
==11549==
==11549== For lists of detected and suppressed errors, rerun with: -s
==11549== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here$
```

## Copying different folders completely (Printing)

```
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ make
gcc -Wall -std=gnu99 main.c -o MMCp -lpthread -lrt
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ valgrind --leak-check=full ./MMCp 3 5 ../testdir/src/xdiff ../tocopy
==11821== Memcheck, a memory error detector
==11821== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==11821== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==11821== Command: ./MMCp 3 5 ../testdir/src/xdiff ../tocopy
==11821==
Copied ../testdir/src/xdiff/xmacros.h to ../tocopy/xmacros.h, 1901 bytes
Copied ../testdir/src/xdiff/xemit.h to ../tocopy/xemit.h, 1204 bytes
Copied ../testdir/src/xdiff/xtypes.h to ../tocopy/xtypes.h, 1529 bytes
Copied ../testdir/src/xdiff/xutils.c to ../tocopy/xutils.c, 9553 bytes
Copied ../testdir/src/xdiff/xutils.h to ../tocopy/xutils.h, 1779 bytes
Copied ../testdir/src/xdiff/xhistogram.c to ../tocopy/xhistogram.c, 9638 bytes
Copied ../testdir/src/xdiff/xemit.c to ../tocopy/xemit.c, 7848 bytes
Copied ../testdir/src/xdiff/xprepare.h to ../tocopy/xprepare.h, 1058 bytes
Copied ../testdir/src/xdiff/xprepare.c to ../tocopy/xprepare.c, 12242 bytes
Copied ../testdir/src/xdiff/README.txt to ../tocopy/README.txt, 647 bytes
Copied ../testdir/src/xdiff/xinclude.h to ../tocopy/xinclude.h, 1641 bytes
Copied ../testdir/src/xdiff/xdiffi.h to ../tocopy/xdiffi.h, 1936 bytes
Copied ../testdir/src/xdiff/xpatience.c to ../tocopy/xpatience.c, 10920 bytes
Copied ../testdir/src/xdiff/xdiff.h to ../tocopy/xdiff.h, 3790 bytes
Copied ../testdir/src/xdiff/COPYING to ../tocopy/COPYING, 26421 bytes
Copied ../testdir/src/xdiff/xdiffi.c to ../tocopy/xdiffi.c, 27698 bytes

-----STATISTICS-----
Consumers: 5 - Buffer Size: 3
Number of Regular Files: 16
Number of FIFO Files: 0
Number of Directories: 0
Number of Other Files: 0
TOTAL BYTES COPIED: 119805
TOTAL TIME: 00:00.118 (min:sec.millis)
==11821==
==11821== HEAP SUMMARY:
==11821==   in use at exit: 0 bytes in 0 blocks
==11821==   total heap usage: 10 allocs, 10 frees, 41,680 bytes allocated
==11821==
==11821== All heap blocks were freed -- no leaks are possible
==11821==
==11821== For lists of detected and suppressed errors, rerun with: -s
==11821== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here$
```



## Copying different folders CTRL+C (Printing)

```
mrguven@mrguven: ~/Desktop/cse344/hw4/hw4test/put_your_codes_here
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ make
gcc -Wall -std=gnu99 main.c -o MWCp -lpthread -lrt
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$ valgrind ./MWCp 12 5 ../testdir/ ../toCopyA
==11896== Memcheck, a memory error detector
==11896== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==11896== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==11896== Command: ./MWCp 12 5 ../testdir/ ../toCopyA
==11896==
Copied ../testdir//lgtn.yml to ../toCopyA/.lgtn.yml, 140 bytes
Copied ../testdir//vintutor.bat to ../toCopyA/vintutor.bat, 1812 bytes
Copied ../testdir//runtime/vin16x16.gif to ../toCopyA/runtime/vin16x16.gif, 166 bytes
Copied ../testdir//runtime/spell/en/en_CA.diff to ../toCopyA/runtime/spell/en/en_CA.diff, 5544 bytes
Copied ../testdir//runtime/spell/en/en_GB.diff to ../toCopyA/runtime/spell/en/en_GB.diff, 5195 bytes
Copied ../testdir//runtime/spell/en/en_US.diff to ../toCopyA/runtime/spell/en/en_US.diff, 5718 bytes
Copied ../testdir//runtime/spell/en/main.aap to ../toCopyA/runtime/spell/en/main.aap, 8650 bytes
Copied ../testdir//runtime/spell/mi/mi_NZ.diff to ../toCopyA/runtime/spell/mi/mi_NZ.diff, 197 bytes
Copied ../testdir//runtime/spell/mi/main.aap to ../toCopyA/runtime/spell/mi/main.aap, 2471 bytes
Copied ../testdir//runtime/spell/zu/zu_ZA.diff to ../toCopyA/runtime/spell/zu/zu_ZA.diff, 174 bytes
Copied ../testdir//runtime/spell/zu/main.aap to ../toCopyA/runtime/spell/zu/main.aap, 2570 bytes
Copied ../testdir//runtime/spell/ca/ca_ES.diff to ../toCopyA/runtime/spell/ca/ca_ES.diff, 1576 bytes
Copied ../testdir//runtime/spell/ca/main.aap to ../toCopyA/runtime/spell/ca/main.aap, 2571 bytes
Copied ../testdir//runtime/spell/sk/sk_SK.diff to ../toCopyA/runtime/spell/sk/sk_SK.diff, 501 bytes
Copied ../testdir//runtime/spell/sk/main.aap to ../toCopyA/runtime/spell/sk/main.aap, 2598 bytes
Copied ../testdir//runtime/spell/tn/tn_ZA.diff to ../toCopyA/runtime/spell/tn/tn_ZA.diff, 381 bytes
Copied ../testdir//runtime/spell/tn/main.aap to ../toCopyA/runtime/spell/tn/main.aap, 2645 bytes
Copied ../testdir//runtime/spell/en/en_AU.diff to ../toCopyA/runtime/spell/en/en_AU.diff, 66901 bytes
Copied ../testdir//runtime/spell/en/en_NZ.diff to ../toCopyA/runtime/spell/en/en_NZ.diff, 66085 bytes
Copied ../testdir//runtime/spell/ro/ro_RO.diff to ../toCopyA/runtime/spell/ro/ro_RO.diff, 828 bytes
Copied ../testdir//runtime/spell/ro/main.aap to ../toCopyA/runtime/spell/ro/main.aap, 3262 bytes
Copied ../testdir//runtime/spell/eo/eo_L3.diff to ../toCopyA/runtime/spell/eo/eo_L3.diff, 0 bytes
Copied ../testdir//runtime/spell/eo/main.aap to ../toCopyA/runtime/spell/eo/main.aap, 2482 bytes
Copied ../testdir//runtime/spell/lt/lt_LT.diff to ../toCopyA/runtime/spell/lt/lt_LT.diff, 0 bytes
Copied ../testdir//runtime/spell/lt/main.aap to ../toCopyA/runtime/spell/lt/main.aap, 2450 bytes
Copied ../testdir//runtime/spell/it/main.aap to ../toCopyA/runtime/spell/it/main.aap, 2476 bytes
Copied ../testdir//runtime/spell/it/it_IT.diff to ../toCopyA/runtime/spell/it/it_IT.diff, 7594 bytes
Copied ../testdir//runtime/spell/es/main.aap to ../toCopyA/runtime/spell/es/main.aap, 3002 bytes
Copied ../testdir//runtime/spell/es/es_ES.diff to ../toCopyA/runtime/spell/es/es_ES.diff, 770 bytes
Copied ../testdir//runtime/spell/hu/hu_HU.diff to ../toCopyA/runtime/spell/hu/hu_HU.diff, 3478 bytes
Copied ../testdir//runtime/spell/hu/main.aap to ../toCopyA/runtime/spell/hu/main.aap, 2628 bytes
Copied ../testdir//runtime/spell/hr/hr_HR.diff to ../toCopyA/runtime/spell/hr/hr_HR.diff, 341 bytes
Copied ../testdir//runtime/spell/hr/main.aap to ../toCopyA/runtime/spell/hr/main.aap, 2589 bytes
Copied ../testdir//runtime/spell/check_locales.vim to ../toCopyA/runtime/spell/check_locales.vim, 463 bytes
Copied ../testdir//runtime/spell/sl/sl_SI.diff to ../toCopyA/runtime/spell/sl/sl_SI.diff, 341 bytes
Copied ../testdir//runtime/spell/sl/main.aap to ../toCopyA/runtime/spell/sl/main.aap, 2590 bytes

Copied ../testdir//runtime/autoload/sqlcomplete.vim to ../toCopyA/runtime/autoload/sqlcomplete.vim, 39188 bytes
Copied ../testdir//runtime/autoload/tar.vim to ../toCopyA/runtime/autoload/tar.vim, 23073 bytes
Copied ../testdir//runtime/autoload/tohtml.vim to ../toCopyA/runtime/autoload/tohtml.vim, 32451 bytes
Copied ../testdir//runtime/autoload/javascriptcomplete.vim to ../toCopyA/runtime/autoload/javascriptcomplete.vim, 27028 bytes
Copied ../testdir//runtime/autoload/getscript.vim to ../toCopyA/runtime/autoload/getscript.vim, 24865 bytes
Copied ../testdir//runtime/autoload/syntaxcomplete.vim to ../toCopyA/runtime/autoload/syntaxcomplete.vim, 31270 bytes
Copied ../testdir//runtime/autoload/netrwSettings.vim to ../toCopyA/runtime/autoload/netrwSettings.vim, 10203 bytes
^CCopied ../testdir//runtime/autoload/htmlcomplete.vim to ../toCopyA/runtime/autoload/htmlcomplete.vim, 25450 bytes
Copied ../testdir//runtime/autoload/xmlcomplete.vim to ../toCopyA/runtime/autoload/xmlcomplete.vim, 14939 bytes
Copied ../testdir//runtime/autoload/context.vim to ../toCopyA/runtime/autoload/context.vim, 5462 bytes
Copied ../testdir//runtime/autoload/clojurecomplete.vim to ../toCopyA/runtime/autoload/clojurecomplete.vim, 8029 bytes
Copied ../testdir//runtime/autoload/decada.vim to ../toCopyA/runtime/autoload/decada.vim, 2998 bytes
Copied ../testdir//runtime/autoload/README.txt to ../toCopyA/runtime/autoload/README.txt, 773 bytes
Copied ../testdir//runtime/autoload/adacomplete.vim to ../toCopyA/runtime/autoload/adacomplete.vim, 3669 bytes
Copied ../testdir//runtime/autoload/astfold.vim to ../toCopyA/runtime/autoload/astfold.vim, 1900 bytes
Copied ../testdir//runtime/autoload/haskellcomplete.vim to ../toCopyA/runtime/autoload/haskellcomplete.vim, 105661 bytes
Copied ../testdir//runtime/autoload/rust.vim to ../toCopyA/runtime/autoload/rust.vim, 10463 bytes
Copied ../testdir//runtime/autoload/cocomplete.vim to ../toCopyA/runtime/autoload/cocomplete.vim, 17857 bytes
Copied ../testdir//runtime/autoload/zip.vim to ../toCopyA/runtime/autoload/zip.vim, 14639 bytes
Copied ../testdir//runtime/autoload/netrw_gitignore.vim to ../toCopyA/runtime/autoload/netrw_gitignore.vim, 3119 bytes
Copied ../testdir//runtime/autoload/csscomplete.vim to ../toCopyA/runtime/autoload/csscomplete.vim, 43248 bytes
Copied ../testdir//runtime/autoload/xml/xhtml10t.vim to ../toCopyA/runtime/autoload/xml/xhtml10t.vim, 56328 bytes
Copied ../testdir//runtime/autoload/netrw.vim to ../toCopyA/runtime/autoload/netrw.vim, 532199 bytes
Copied ../testdir//runtime/autoload/phpcomplete.vim to ../toCopyA/runtime/autoload/phpcomplete.vim, 354119 bytes

-----STATISTICS-----
Consumers: 5 - Buffer Size: 12
Number of Regular Files: 616
Number of FIFO Files: 0
Number of Directories: 85
Number of Other Files: 0
TOTAL BYTES COPIED: 7424740
TOTAL TIME: 00:01.521 (min:sec.millis)
==11896==
==11896== HEAP SUMMARY:
==11896==    in use at exit: 0 bytes in 0 blocks
==11896==   total heap usage: 95 allocs, 95 frees, 2,849,544 bytes allocated
==11896==
==11896== All heap blocks were freed -- no leaks are possible
==11896==
==11896== For lists of detected and suppressed errors, rerun with: -s
==11896== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
mrguven@mrguven:~/Desktop/cse344/hw4/hw4test/put_your_codes_here$
```