1) development environment and how to build dependencies
+) ubuntu-18.04.4-desktop-amd64.iso (Linux OS)
+) gcc/g++ 7.5.0 (with std::tr1 support for function, bind, shared_from_this, ...)
+) pthread (multi thread)
+) glog (debug logging)
+) gtest (googletest unit test)
+) valgrind (memory leak)
+) exprtk (c++ mathematical expression library)

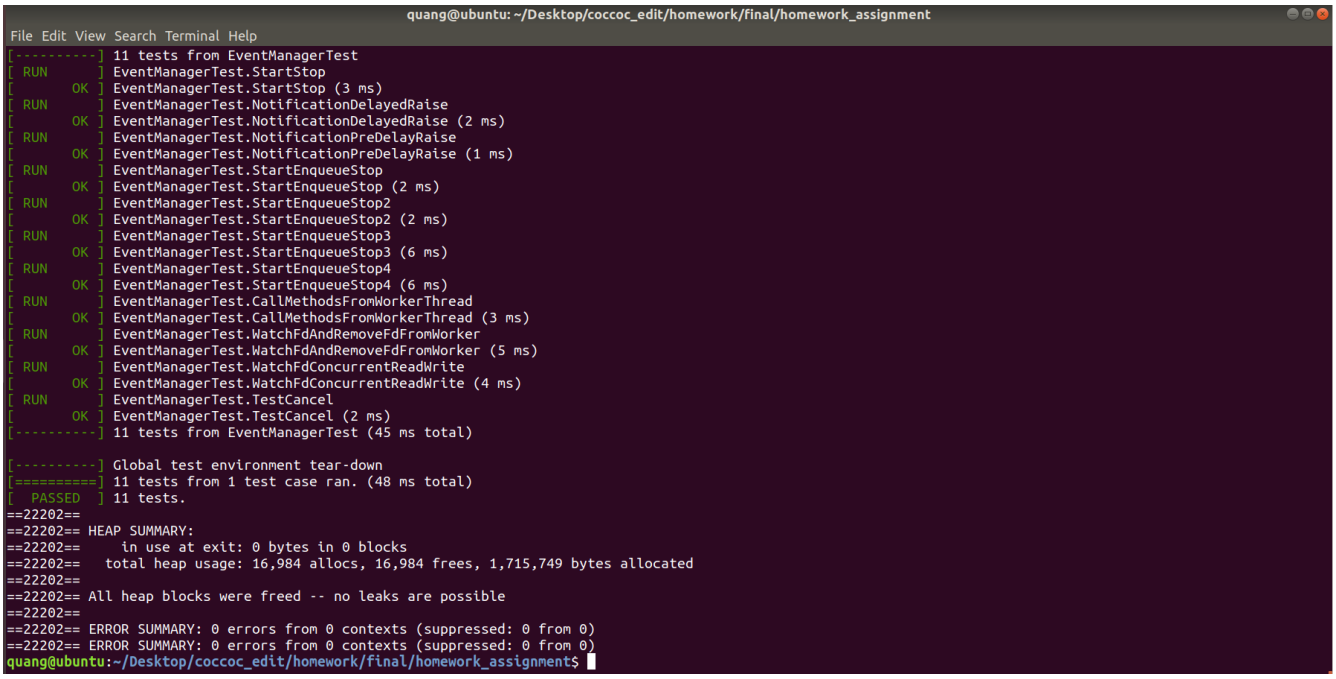
a) step 1: update ubuntu with command line:
    sudo apt update
b) step 2: install gcc/g++ with command line:
    sudo apt install gcc g++
c) step 3: libpthread.so is available in ubuntu 18.04.4 with path "/usr/lib/x86_64-linux-gnu"
d) step 4: install glog with command line:
    sudo apt install autoconf libtool make
    tar xvf glog-0.4.0.tar.gz
    cd glog-0.4.0/
    ./autogen.sh
    ./configure --prefix=$HOME/CocCoc/glog-0.4.0
    make
    make install
e) step 5: install gtest with command line:
    sudo apt install cmake
    tar xvf googletest-release-1.8.1.tar.gz
    cd googletest-release-1.8.1/
    mkdir mybuild
    cd mybuild/
    cmake ../ -DCMAKE_INSTALL_PREFIX=$HOME/CocCoc/googletest-release-1.8.1
    make
    make install
f) step 6: install valgrind with command line:
    sudo apt install valgrind

2) how to build unit test (googletest) using makefile
+ step 1: clean all with command line as follow:
    make clean
+ step 2: build eventmanager test with command line as follow:
    make eventmanager_test
+ step 3: build iobuffer test with command line as follow:
    make iobuffer_test
+ step 4: build tcp test with command line as follow:
    make tcp_test

3) how to run unit test (googletest) and memory leak (valgrind) using makefile

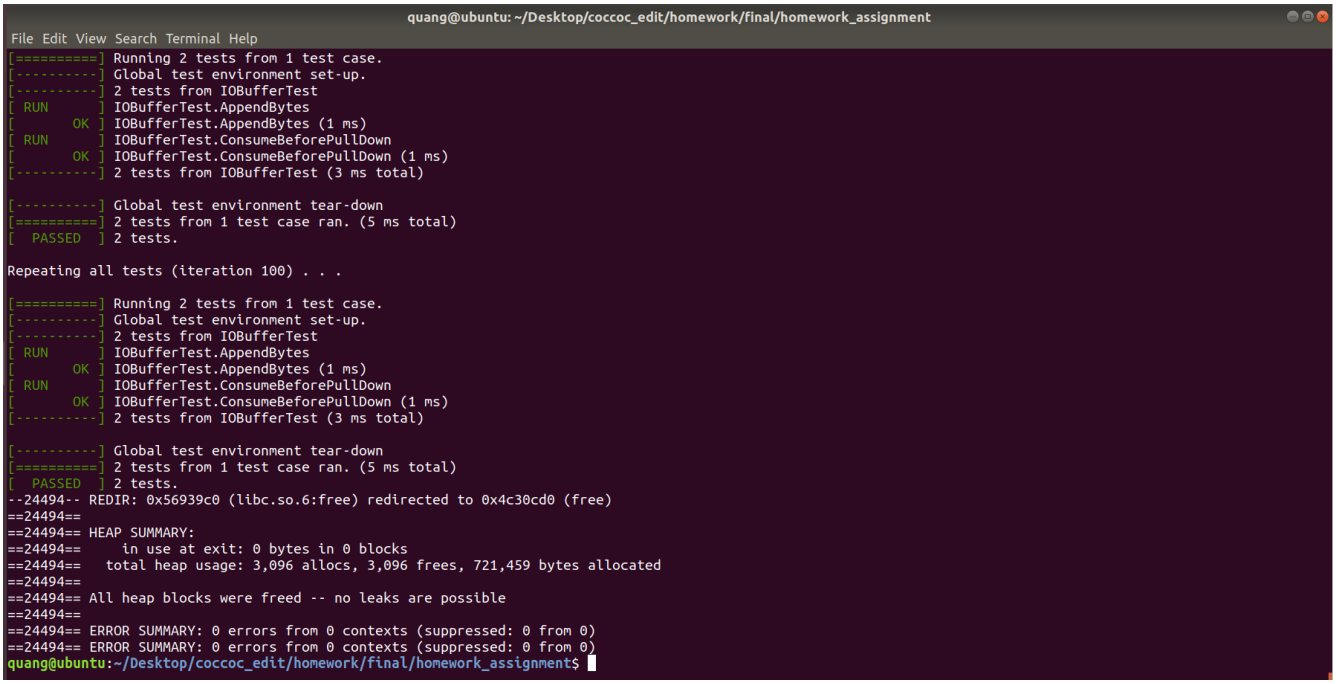
+ step 1: run eventmanager test with command line as follow:

make eventmanager_valgrind

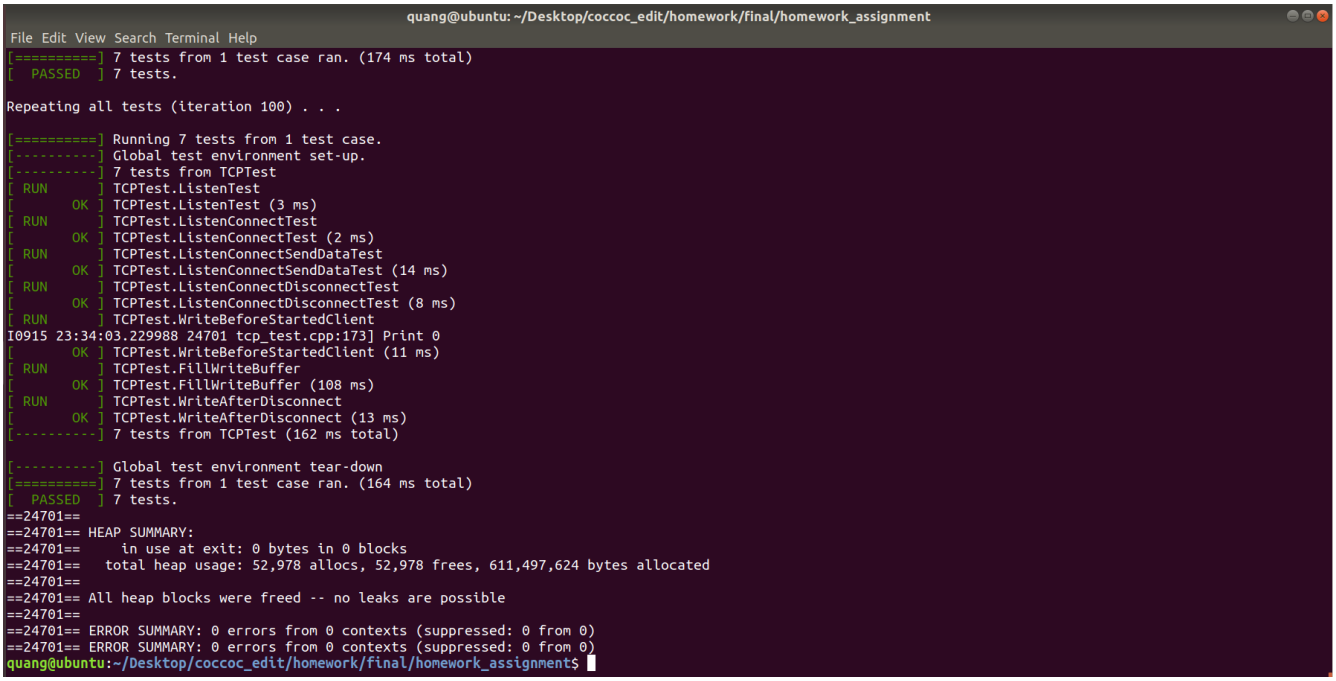


+ step 2: run iobuffer test with command line as follow:

make iobuffer_valgrind



+ step 3: rum tcp test with command line as follow:
make tcp_valgrind

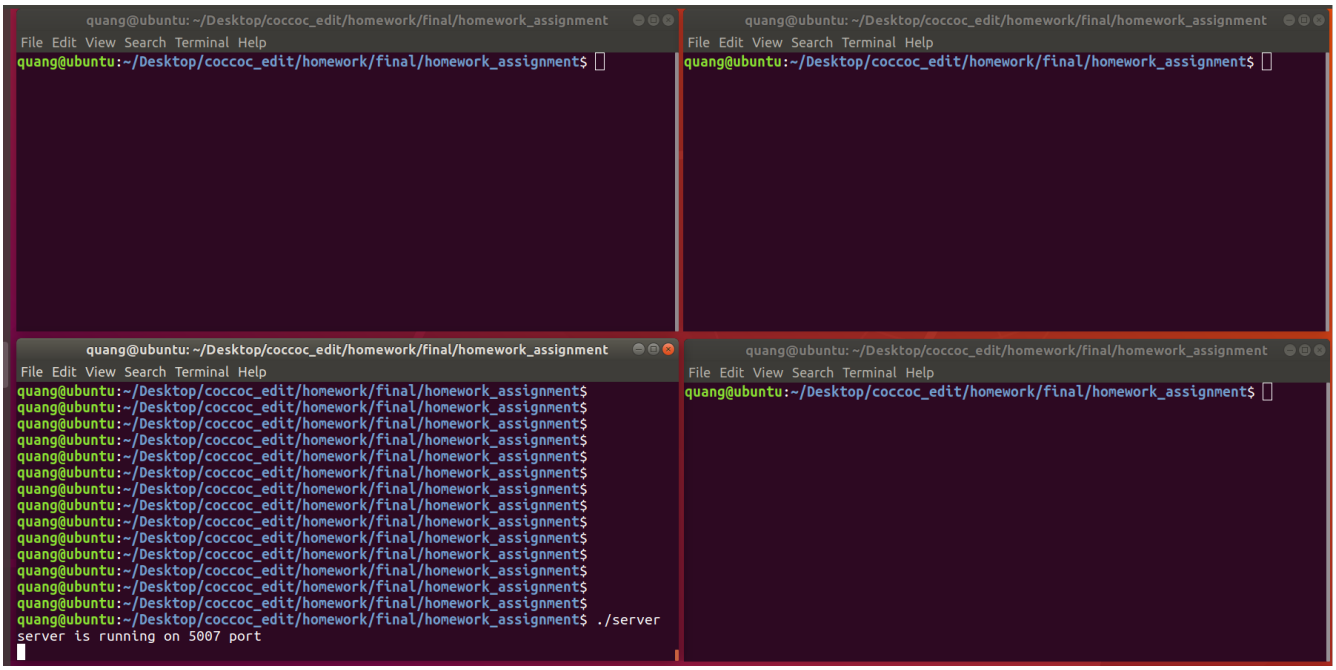


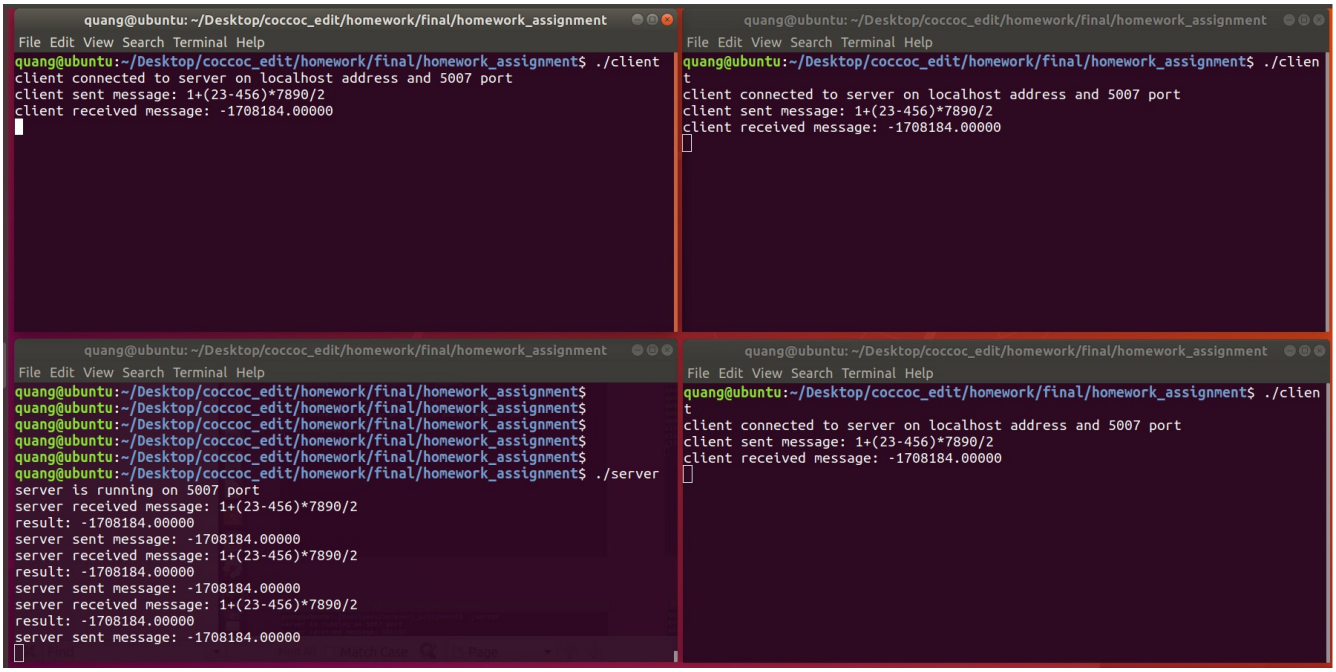
4) how to build socket (server and client) using makefile
+ step 1: build server with command line as follow:
make server
+ step 2: build client with command line as follow:
make client

5) how to test communication between server and client via tcp socket

+ Server is running on 5007 port: run command line as follow:

./server



+ Client connected to server on localhost address and 5007 port: run command line as follow:

./client