NAZARBAYEV UNIVERSITY
SCHOOL OF SCIENCE AND TECHNOLOGY

Project Assignment #3

ROBT305 – EMBEDDED SYSTEMS

Fall Semester 2018

PHTREADS PROGRAMMING PRACTICE.
IMU SENSOR GUI DEVELOPMENT USING QT

DUE TIME AND DATE

- **Class presentation and report uploaded to Moodle on Tuesday 30 October.**

LEVEL OF COLLABORATION ALLOWED

You will be working in groups of two students

DELIVERABLES REQUIRED

You group is required to prepare and submit to Moodle system an MS Word report with Linux terminal screenshots for all major tasks and steps from this assignment. In addition you will present your assignment in class.

REFERENCES

- **Abraham Silberschatz, Peter B. Gatvin, Greg Gagne, "Operating System Concepts", 9th International edition, Wiley, 2014 (available in library) DO NOT USE PDFs of the textbook US editions from Internet - the task numbers are different in these textbook versions.**
- **Derek Molloy,"Exploring BeagleBone. Tools and Techniques for Building with Embedded Linux", Wiley, 2015 (available in Moodle and in library)**
- **G. Lazar, R. Penea, Mastering Qt5, Pact, 2016 (available in Moodle)**

INTRODUCTION

In this assignment you will develop a multithreaded IMU sensor orientation visualization GUI.

ASSIGNMENT DETAILS

TASK 1. POSIX THREADS PROGRAMMING AND SYNCHRONIZATION (70% OF THE TOTAL GRADE)

1. **(60 points)** On your laptops/PCs complete exercises #4.16, 4.17, 4.21 and Programming Project 1 (Sudoku Solution Validator) on pages 193-196, exercise #6.33 on page 300 of the Operating System Concepts textbook by A. Silberschatz, P. Galvin, G. Gagne (available in library).

2. **(10 points)** Programming projects #1 – 3 on page 301 of the Operating System Concepts textbook by A. Silberschatz, P. Galvin, G. Gagne.

You are given example solutions for these three projects. Run the solution codes and verify its correctness in accordance to the project description in the textbook. Write a report with a detailed description of the project code and its operation including screenshots.

In the above tasks please provide screenshots of your programs (code and run) in the report.

You can refer to the POSIX threads programming tutorial with exercises https://computing.llnl.gov/tutorials/pthreads for additional practice on PThread programming (available in Moodle)

TASK 2. THREADED IMU GUI DEVELOPMENT (20% OF THE TOTAL GRADE)

In this task you will develop a client/server application to visualize an IMU sensor orientation on your laptop/PC. The BBB with the interfaced IMU sensor acts as a server and sends processed orientation data to your laptop/PC (client). The client runs a Qt GUI to visualize the IMU sensor 3D orientation. Please implement the following:

1. Convert your GUI code from Project 2 to a multithreaded application using PThreads library as follows:

   On the server side (BBB):

   - Thread 1 reads the IMU sensor data and passes it to Thread 2. Before passing the raw sensor measurement data you have to convert it to physical values.
   - Thread 2 runs the open-source Magwick IMU sensor measurement fusion algorithm (from Assignment #2) and sends the quaternion estimates to Thread 3.
   - Thread 3 implements the server socket communication and sends the data to the socket for reading by the client side.

   Use shared variables (protected by mutexes if needed) to pass the data from one thread to another.

   You may follow the online C++ GUI with Qt Tutorial videos #28 to #35 (if deemed necessary) from https://www.youtube.com/watch?v=JaGqGhRW5Ks for learning how to work with QThread class in the Qt.

GRADING CRITERIA

Demonstration and grading of your working PThread programs (Task 1) and BBB&IMU setups (Task 2) setups and reports (with questions/answers) will be done during the class lab session on Tuesday 30 October.

The total grade for all implemented assignments as described will constitute 90%. The bonus grade will be awarded to the groups showing outstanding coding experience in the form of complexity of modifications in the Task 1 codes, e.g. implementation of the tasks using QThreads in the Qt Creator with some GUI designs, additional features, completeness of the project reports, etc.

Please prepare a detailed report with program code and submit it to the project folder in Moodle by the end of Tuesday 30 October.

This project evaluation will be done using individual grading depending on the level of participation and understanding of the project assignments.

Late submission penalty – 10% per day

@Almas Shintemirov email: ashintemirov@nu.edu.kz