Laboratory exercise 6

Building projects on Linux using CMake

Name: Dunja Šmigovec **JMBAG:** 0036516239

Preparation and helpful instructions

- Review the lecture slides about compiling and linking C++ code on linux.
- Look for some ROS projects that use external shared libraries and see how they have setup the CMakeLists.txt file.
- For the first task, you will need to install the Boost library. On Ubuntu 20.04, we recommend installing version 1.71 by invoking the command `sudo apt install libboost1.71-all-dev`, but any other version should suffice.
- For the second task, you will need to figure out CMake commands find_package and include_directories.

Assignments

Task 1: Linear algebra using the Boost library

In this assignment you will write and build a program that carries out simple linear algebra operations using the Basic Linear Algebra Library (uBLAS) which is a part of Boost. Boost is a set of widely used free peer-reviewed portable C++ source libraries that can help you speed up development by reducing the *reinvention-of-the-wheel*. You can access the uBLAS documentation here. Do not forget to import relevant uBLAS header files in your source file(s). Write the `CMakeLists.txt` file, name the project and the executable file `firstname_lastname`. Build the program with CMake.

a) Inside the `main()` function using uBLAS define two matrices `m1` and `m2` with dimensions 5 × 5 and a vector `v` with dimension 5. Fill the matrix `m1` such that each row contains the first 5 digits of your JMBAG number. Generate matrix `m2` by summing `m1` with the identity matrix of the same size. Fill the vector `v` with the last 5 digits of your JMBAG number.

b) Multiply the matrix `m2` and the vector `v` and write the result on the standard output. Paste the output in the following text box.

```
[5](70,75,71,72,78)
```

c) Multiply the vector `v` with its transpose and write the result on the standard output. Paste the output in the following text box.

```
[1](131)
```

d) Calculate the determinant of the matrix `m2` and write it on the standard output. Paste the output in the following text box.

```
15
```

e) Calculate the inverse of the matrix `m2` and write it on the standard output. Paste the output in the following text box.

```
[5,5]((1,0,-0.2,-0.4,-0.333333),(0,1,-0.2,-0.4,-0.333333),(0,0,0.8,-0.4,-0.333333),(0,0,-0.2,0.6,-0.333333
```

Task 2: Face detection using the OpenCV library

In this assignment you will build a face detector using OpenCV, which is an open source computer vision and machine learning software library. Unpack `faceDetection.zip` which contains the source code and parameters files for this task. Your job is to write the `CMakeLists.txt` and compile the program with CMake.

a) Check if you have under ROS the `objdetect, highgui` and `imgproc` OpenCV libraries installed (required for the program). In which folder are these shared libraries situated?

```
/usr/include/opencv4/opencv2/
```

b) In the same folder as `objecDetection.cpp` copy any `.jpg` image with a human face.

c) Write the `CMakeLists.txt` file from scratch, build the program with CMake and run the executable file. If a face was detected program will create `face_detection.jpg` image with the detected face and eyes. In the following text field write every command that you used in the process.

```
mkdir build
cd build
cmake ..
make
./objectDetection
```

Exercise submission

Create a zip archive containing **this pdf with the filled out answers** and **all other exercise files**: CMakeLists.txt and the source file(s) for the first task, CMakeLists.txt and `face_detection.jpg` for the second task. Upload and submit on Moodle.