Laboratory exercise 4

Building projects on Linux using CMake

Name: Sérgio Miguel Rosa Estêvão **JMBAG:** 0036563035

Preparation and helpful instructions

- Review the lecture slides about compiling and linking C++ code on linux.
- For the first task, you will need to install the Eigen library. Install the Eigen library by executing the command `sudo apt install libeigen3-dev`.
- For the second task, you will need to figure out CMake commands find_package and include_directories.

Assignments

Task 1: Linear algebra using the Eigen library

In this assignment you will write and build a program that carries out simple linear algebra operations using the Eigen library. Eigen is a set of commonly used free C++ source libraries provide manipulation of matrices, vectors, and other structures commonly used in solving scientific and engineering problems. You can access the Eigen documentation here. For brief instructions on how to install and use Eigen, see here.

Do not forget to import relevant Eigen header files in your source file(s). Write your code in a C++ file named `firstname_lastname.cpp`. 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 Eigen 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.

```
49 46 43 46 48
```

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

```
36 18  0 18 30
18  9  0  9 15
```

d) Calculate the sum of matrices `m1` and `m2` and write it on the standard output. Paste the output in the following text box.

```
1  0  6 12 10
 0  1  6 12 10
```

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

```
1        0      -0.2      -0.4 -0.333333
0        1      -0.2      -0.4 -0.333333
```

f) Transpose matrix `m2` and write it on the standard output. Paste the output in the following text box.

1 0 0 0 00 1 0 0 03 3 4 3 36 6 6 7 65 5 5 5 6

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 the `objdetect, highgui` and `imgproc` OpenCV libraries installed (required for the program). They should have been installed within your ROS installation. What is the location of the folder where these shared libraries are placed?

/usr/lib/x86_64-linux-gnu

b) Copy any `.jpg` image with a human face in the same folder as your `objecDetection.cpp` source file. Name the image `face.jpg`.

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.

cmake .
make
./ObjectDetection

Exercise submission

Create a zip archive containing **this pdf with the filled out answers** and **all other exercise files**: CMake-Lists.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.