# Testing Impact Study

## Task A\_1 (15 minutes)

1. Describe how did you solve the assignment?

In order to solve the assignment first I looked through the program to understand what it is supposed to do, and after that I started to implement to function. To implement the method I just iterate through every element of the “lstEmotion” list and compare every element from the list with the input emotion.

1. Did you use “manual” testing during implementation?

In the application there isn’t any user interface or so called “main” to be run, so in order to check if the function execute correctly, I wrote a small test to see if the function was run correctly

1. If yes, with what date have you test it? – No answer
2. How many executions before you declared “satisfied” with the implementation?

There were 3 execution before I was declared with the final method. On every execution I was looking for different emotions to see if anything worked as it is supposed to, and after I saw that the function was working just fine I declared myself “satisfied” with the method that I wrote.

## Task A\_2 (25 minutes)

Describe how you design your test cases, what techniques have you used, what strategies.

The test cases were designed using a white-box testing strategy. I looked at the code and figure out every path that the algorithm was supposed to take and created test cases for every possible input data. By doing so 3 test cases were created: one for the possibility of only one predominant emotion, one for multiple predominant emotions and one for no emotions at all. By those 3 tests all the possible path in the algorithm were checked.

## Task A\_3 (15 minutes)

1. Describe how did you use the provided test cases to “check” for your algorithm correctness? I looked at the tests and from them I understand better what I am supposed to do in the method that I was supposed to implement
2. How many modifications had you to make before all test cases to be “pass”? No modifications were required. All the tests “passed” from the first run.