Structured Programming – Assessment Task 3

# Structures, nested structures, arrays of structures, streams.

# **Description**

1. Create the structure that describes any object of physical or virtual world (**not a student!**)**.** The fields of the structure should be of different data types: *int, float, char, char\* etc.* At least one of the fields should be of other structure type (nested structure), e.g.:

struct s\_address **{**

char street**[**30**];**

int building\_number**;**

int flat\_number**;**

**};**

struct student **{**

char first\_name**[**30**];**

char last\_name**[**30**];**

struct s\_address address**;**

**};**

1. Create the ***array*** of this structure.

For maximum (excellent) grade:

1. Fill the data of the elements in the array by ***reading this data from file***. The preference will be given for implementations with other then ***fscanf()*** reading from file techniques. Minimum quantity of elements in the array – 10.
2. Perform modification of data of any elements in the array.
3. Output the resulting array of structures to the file.

For lower grades (8 and lower):

1. Fill the data of the elements in the array by “hardcoding” the data inside the program.
2. Perform modification of data of the elements in the array.
3. Print all the data to the screen.

# Requirements

1. Use **functions and procedures** for **all** operations and calculations.
2. The entire code should be properly formatted.
3. Code should follow the 'Hungarian notation' rules.
4. Code should be properly documented.
5. GOTO statement is not allowed.
6. The deadline is January 6th.

# References

1. <https://users.ece.cmu.edu/~eno/coding/CCodingStandard.html>
2. <http://www.doxygen.nl/manual/docblocks.html>
3. <https://developer.lsst.io/cpp/api-docs.html#documenting-c-code>