**Continuous Assessment 2:**

**Student Records Management system using CLASSES**

**Deadline**:

2nd May 2021

This CA is worth **30% of your overall module mark**. You may be asked to do a small demo of your system.

**Submission**:

Submit all your Python code as through Brightspace.

**Task**:

You are asked to develop a small college management system to illustrate your understanding of the main object-oriented concepts.

Your system should keep track of **Students**. Each student has a **student ID**, **name**, **email address** and **list of current modules** they are taking. Each student can enrol in up to 5 modules max.

Each **Module** has an **unique code**, a **name**, a **number of ECTS credits**. Each module also has a max capacity, and once that is reached no more students can be enrolled until somebody unenrolls first.

You’ll need to keep track of, and update, what modules are students enrolled in.

Some of the functionality your system should provide is:

* Print and update details about the students
* Print and update details about the modules
* Search for a student using different parameters (e.g. by email or student ID)
* Enrol and unenroll a student from a module
* Create and delete students and modules

Make sure you include any relevant error checking and handle unexpected input.

**MARKING GUIDELINES**

*The following is only a guide to the marking scheme and could be slightly modified.*

|  |  |
| --- | --- |
| Design classes, use of OOP principals, etc. | (15 marks) |
| Print details about modules and students | (15 marks) |
| Add/remove modules and students | (15 marks) |
| Enroll/unenroll a student on a module | (15 marks) |
| Search for a module and user (using different parameters, not only id) | (15 marks) |
| Overall style, comments, functions; also marks given for menu system | (25 marks) |
| **TOTAL** | (100 marks) |