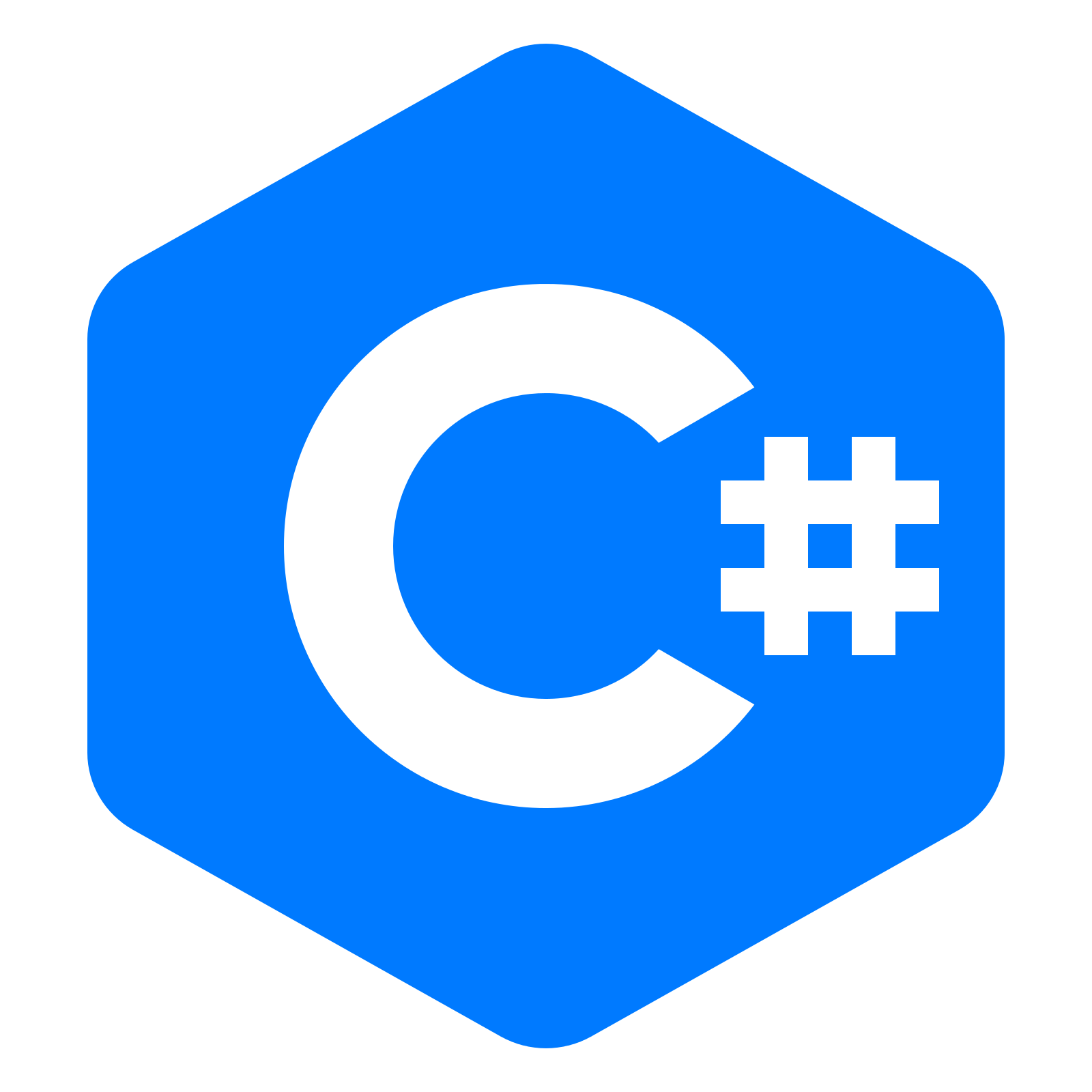
UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

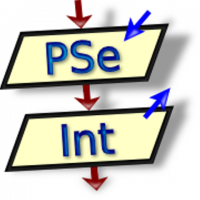
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

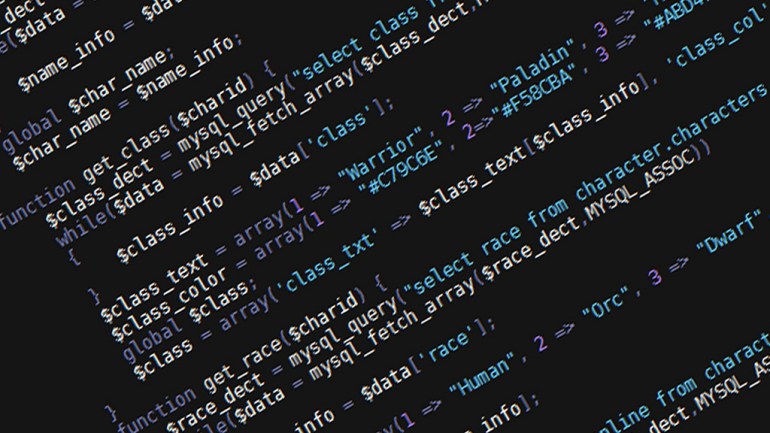
**OBJECT ORIENTED PROGRAMMING**











**Student:** Llumiquinga Moreno Jerson.

E-mail: [jsllumiquinga1@espe.edu.ec](mailto:jsllumiquinga1@espe.edu.ec)

#Cell: 0980460057

**Teacher:** Ing. Edison Lascano.

E-mail: [jelascano@espe.edu.ec](mailto:jelascano@espe.edu.ec)

#Cell: 0961195050

**NRC :** 14539

**Semester**:#02

**MAY24 - SEP24**

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

OBJECT ORIENTED PROGRAMMING

**Name**: Llumiquinga M. Jerson **Date:** 08/07/2024

**Grade Registration System**

**METHODS:**

1. **Administrator**
   1. calculateGPA(List<Double> grades)
   2. loadData(String fileName, Type type)
   3. evaluateStudent(Student student)
2. **Course**
   1. addStudent(Student student)
   2. getStudentById(String studentId)
   3. setSubject(Subject subject)
   4. getProfessor()
3. **Grade**
   1. Grade(String studentId, String subjectCode, double grade)
   2. setSubjectCode(String subjectCode)
   3. setGrade(double grade)
4. **Professor**
   1. Professor(String id, String name, String department)
   2. setId(String id)
   3. setName(String name)
   4. setDepartment(String department)
5. **Student**
   1. Student(String id, String name, int age, String major)
   2. addGrade(double grade)
   3. updateGrade(int index, double grade)
   4. setId(String id)
   5. setAge(int age)

**TEST CASES:**

* 1. **calculateGPA**

input1: float

output: float (GPA)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | GRADE 1 | GRADE 2 | GRADE 3 | GPA |
| 1.1.1 | 10 | 9 | 7 | 8.66666 |
| 1.1.2 | 5 | 10 | 9 | 8 |
| 1.1.3 | 0 | 1.5 | 7.5 | 3 |

* 1. **loadData**

input: String fileName

output: void, replacing with the JSON file

input Data: id: 1, Name. Jerson Stiven Llumiquinga Moreno, age: 19, major: math, grades: 10,9,7

output:

{"id": "1", "name": "Jerson Stiven Llumiquinga Moreno", "age": 19, "major": "Math", "GPA": 8.66666, "grades": [10.0, 9.0, 7.0] }

* 1. **evaluateStudent**

input: GPA

output: String

input Data: [10.0, 9.0, 7.0]

outputData: “Evaluation: The student has passed”.

**2.1 addStudent(Student student)**

Input: Student student

Output: Void. The student is added to the students list.

**2.2 getStudentById(String studentId)**

Input: String

Output: Student

**2.3 setSubject(Subject subject)**

Input: subject

Output: Void (Establish a new subject for the course)

**2.4 getProfessor()**

Input: Null

Output: Professor (A professor of course)

**3.1 Grade(String studentId, String subjectCode, double grade)**

Input: String, String, Double

Output: Void (initialize studentId, subjectCode, and grade)

**3.2 setSubjectCode(String subjectCode)**

Input: String

Output: Void (Set a new value for subjectCode)

**3.3 setGrade(double grade)**

Input: double

Output: Void (Set a new value for Grade)

* 1. **Professor (String id, String name, String department)**

Input: String id, String name, String department

Output: (Void). Attributes are initialized

* 1. **setId(String id)**

input: String

Output: (Void). New value for id

* 1. **setName(String name)**

Input: String

Output: (Void). New value for name.

* 1. **setDepartment(String department)**

Input: String

Output: (Void). New value for department

**5.1 Student(String id, String name, int age, String major)**

Input: String (Id), String (name), int (age), String (major)

Output: Attributes are initialized

* 1. **addGrade(double grade)**

Input: double (grade)

Output: (Void). The grade is added to the major list and the gpa is updated.

* 1. **updateGrade(int index, double grade)**

Input: int (index)

Output: (Void) The grade in the grades list is updated at the specified index and the gpa is updated.

* 1. **setId(String id)**

Input: String id

Output: a new value is set for id.

**5.5 setAge(int age)**

Input: int age

Output: A new age is for the student.