Open University of Sri Lanka

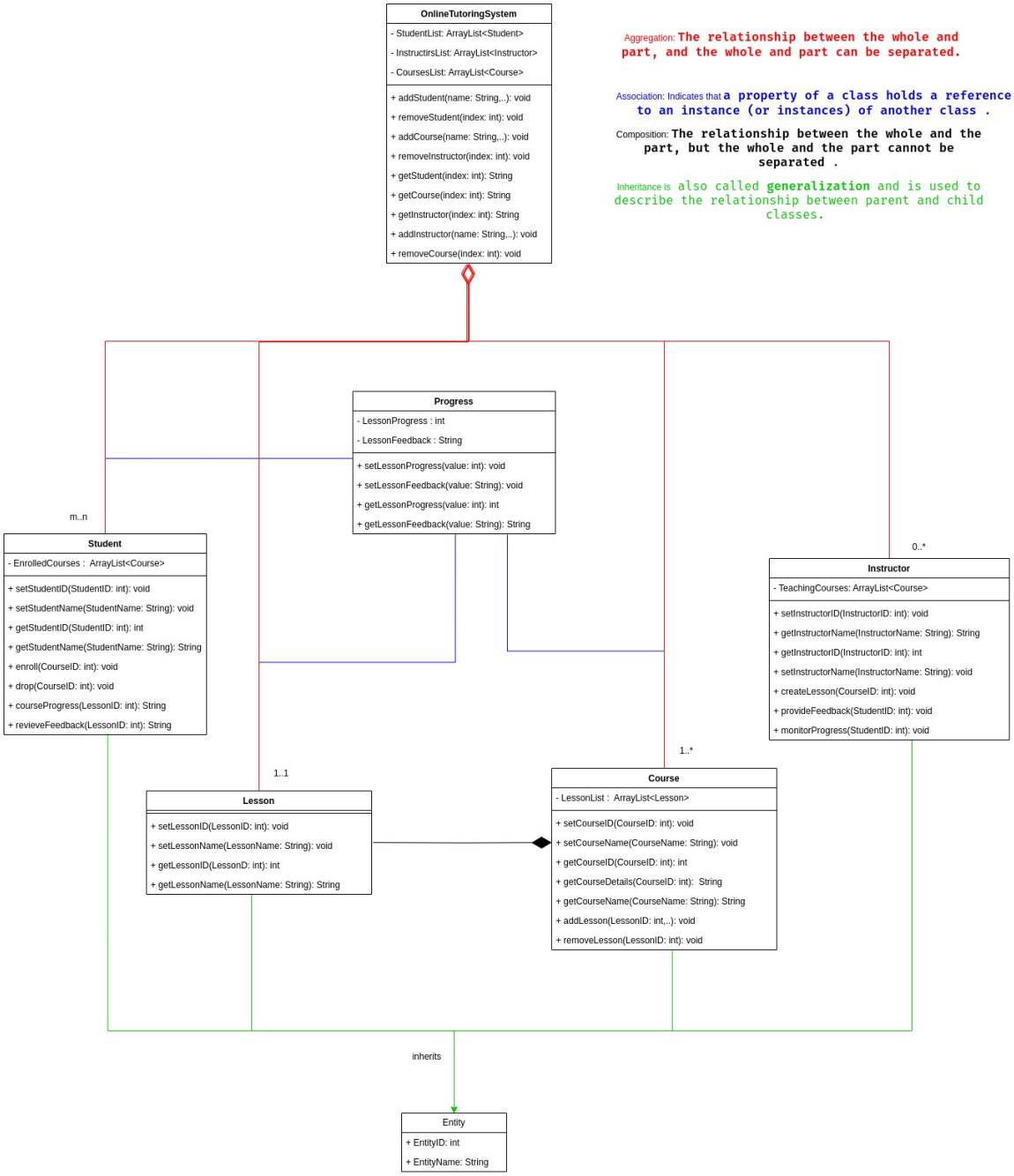
Department of Electrical and Computer Engineering

Faculty of Engineering Technology

EEI4362- Object Oriented Design

id: s92062851 | reg#: 321422851

Q1



Q2

Q2.1

Q2.1.1

```java
import java.util.Arrays;
public class ExampleClass {
    public static int calculateSum(int[] numbers) {
        return Arrays.stream(numbers).parallel().sum();
    }
}
```

In this scenrio ‘Arrays’ class is used to sum this ‘int’ array with built-in ‘sum’ method. To deal with efficiency i utilized Java’s ‘parallel’ method, which would access array elements in parallel, leveraging multiple threads for computation. This approach is more suitable for processing arrays with large elements since it divides summation process among multiple threads.

Q2.1.1

Importance :

In general as the modules and functionalities in a program increases, it’s more vulnarable for being very slow and insecure. This problem would later turn out to be a big issue since they’d require high resource usage, and bad user experience. Thus it’s necessary to address these issues by following best practices.

Benefits :

- Improves overall performance of the program
- Helps reduce resource usage
- Helps organize and struture the codebase

Trade-offs:

- Time consuming
- Increased complexity of the code
- Decreases code readability

Q2.2

`Interface` is a way of achieving abstration to provide the subclasses who implement the interface with a contract in which the behaviours and attributes of the class are spcified. A class that implements an interface must define all signatures before instantiation.

In the scenario there is an ‘Interface’ called ‘Drawable’ which has a predfined method called ‘draw()’ without any declaration.

Both classes ‘Circle’ and ‘Square’ who implement ‘Drawable’ also define the method ‘draw()’ since it’s necessary.

Q3

User Registration Process: