

Marking

2 marks for each question.

Common Mistakes

`count()` — used to count **the occurrences of a specific element** in a collection, not to count the total number of elements (for that, we use `size()`).

`if-else` — not in OCL either use `implies` or `if-then-else`

Using `!=` instead of `<>`

Using `&&` — not in OCL use `and` instead

Solution :

1. Departments should not be null:

```
context University inv:  
    self.departments->forall(d | d <> null)
```

OR

```
context Department inv:  
    self.university <> null
```

2. A university must have at least one department:

```
context University inv:  
    self.departments->size() >= 1
```

```
context University inv:  
    self.departments->notEmpty()
```

3. At least one student named Ali in university:

```
context University inv:  
    self.departments.students->exists(s | s.name = 'Ali')
```

```
context University inv:  
    self.departments->collect(d | d.students)->exists(s | s.name = 'Ali')
```

4. Average grade is always greater than 4:

```
context Student inv:
  self.grades->collect(g | g.value)->avg() > 4
```

```
context Student inv:
  self.avgGrade > 4
```

OR

```
self.avgGrade() > 4    // if function is defined
```

5. A student cannot follow more than 3 courses at a time:

```
context Student inv:
  self.courses->size() <= 3
```

6. If title is 'full', he teaches at least 3 courses:

```
context Instructor inv:
  self.title = 'full' implies self.courses->size() >= 3
```

OR

```
self.title = Title::full implies self.courses->size() >= 3
```

7. Wage of all instructors in a department is at least 1 lac

```
context Department inv:
  self.instructors->forall(i | i.wage >= 100000)
```

OR

```
Context Instructor inv:
Self.wage > 100000
```