

## BÀI TẬP TUẦN 4

**Nội bài: Nén thư mục *src* của project thành file nén có tên theo định dạng *HoTenSV\_MSSV\_X.zip***

Xử lý điểm sinh viên

Thông tin về điểm của một môn học được lưu trong một file văn bản có tên file là Mã môn học ghép cùng mã học kỳ (Mã môn học\_Mã học kỳ) với định dạng như sau:

- Một dòng với mã môn học (có chữ "SubjectID")
- Một dòng với tên môn học (với tiêu đề "Subject")
- Một dòng "F" cho biết hệ số các điểm quá trình và cuối kỳ(Theo %)
- Một dòng với mã học kỳ (có chữ "Semester")
- Một dòng "StudentCount" cho biết số sinh viên đăng kí môn học (tương ứng số dòng còn lại trong file)
- Sau đó là các dòng "S" với thông tin mã SV, Họ-Tên đệm, Tên và các điểm thành phần (theo thang 10), Điểm môn học (Thang điểm chữ theo quy định)

Điểm chữ	A	B	C	D	F
Điểm số	8.5-10	7.0 - Dưới 8.4	5.5 - Dưới 7.0	4.0 - Dưới 5.5	0 - Dưới 4.0

Kí tự phân cách là "|"

Ví dụ mẫu file môn học IT3650 có tên IT3650\_20141.txt như sau:

```
SubjectID|IT3650
Subject|Java Programming
F|30|70
Semester|20141
StudentCount|26
S|20101160|NGUYEN VAN      |AN      | 8.5    | 7.0    | B |
S|20191182|TRAN QUOC        |BINH    | 10.0   | 9.0    | A |
S|20111191|LE BANG                   |CHAN    | 5.5    | 4.5    | D |
S|20101216|NGUYEN VAN                |DUONG   | 8.5    | 8.5    | A |
```

Điểm của môn học được thống kê và ghi trên file văn bản có tên định dạng Mã môn học\_Mã học kỳ\_rp. Định dạng nội dung của file này như ví dụ dưới đây

IT3650\_20141\_rp.txt

```
The student with the highest mark is: TRAN QUOC BINH
The student with the lowest mark is: LE BANG CHAN
The average mark is: 7.51
```

A histogram of the subject IT3650 is:

A: \*\*  
B: \*  
C:  
D: \*  
F:

Trong đó, số dấu \* là số sinh viên đạt điểm chữ tương ứng.

Nếu có nhiều sinh viên có cùng điểm cao nhất, chỉ cần đưa ra tên sinh viên đầu tiên. Tương tự với điểm thấp nhất.

**Câu 1.** Định nghĩa lớp **StudentScore** để xử lý thông tin về điểm của một sinh viên trong lớp học. Các thuộc tính cần có:

- Họ và tên đệm
- Tên
- MSSV
- Điểm quá trình
- Điểm cuối kỳ

Ngoài các phương thức khởi tạo, getter và setter cần thiết, lớp cần có thêm 2 phương thức public:

- Phương thức getMark() tính điểm chữ: đối số là hệ số điểm quá trình, giá trị trả về là một ký tự điểm chữ
- Phương thức getMarkLine() trả về một xâu có định dạng tương tự như dưới đây để ghi vào file văn bản chứa kết quả môn học

```
S|20101160|NGUYEN VAN          |AN      | 8.5   | 7.0   | B   |
```

**Câu 2.** Định nghĩa lớp **Subject** để xử lý thông tin về môn học trong chương trình đào tạo. Các thuộc tính cần có:

- Mã môn học
- Tên môn học
- Hệ số điểm quá trình
- Hệ số điểm cuối kỳ (không được nhập bởi người dùng mà tính qua hệ số điểm quá trình)

Viết các phương thức khởi tạo, getter và setter cần thiết

**Câu 3.** Khai báo một lớp có tên **ScoreBoard** để xử lý thông tin điểm của môn học trong một học kỳ. Lớp này gồm có các thuộc tính:

- Một đối tượng thuộc lớp Subject
- Một mảng các đối tượng thuộc lớp StudentScore
- Mã học kỳ
- Số lượng sinh viên

Các phương thức của lớp bao gồm:

- Phương thức khởi tạo thực hiện tạo file lưu thông tin điểm của môn học nếu file chưa tồn tại. Ngược lại, nếu file đã tồn tại đọc thông tin vào các thuộc tính đã mô tả ở trên.
- Các phương thức setter để thiết lập giá trị cho các thuộc tính, trừ mảng kiểu StudentScore
- Phương thức thêm thông tin điểm của một sinh viên. Nếu sinh viên chưa có điểm (chưa có trong mảng) thì thêm mới vào mảng. Nếu sinh viên đã có trong mảng, hỏi lại người dùng có cập nhật điểm mới không và xử lý theo quyết định của người dùng.
- Phương thức xóa thông tin điểm một sinh viên khỏi mảng. Nếu xóa được trả lại giá trị true, ngược lại trả lại giá trị false.
- Phương thức để cập nhật toàn bộ thông tin từ mảng kiểu StudentScore vào file văn bản. Lưu ý khi cập nhật vào file cần đảm bảo đúng thứ tự sinh viên theo tên.
- Phương thức tạo mới file báo cáo có nội dung theo cấu trúc đã mô tả. Nếu file báo cáo đã tồn tại thì cập nhật nội dung mới.
- Các phương thức getter nếu cần thiết.

**Câu 4.** Xây dựng lớp **ScoreManagement** để hiển thị menu sau và thực hiện theo các chức năng tương ứng:

```
Learning Management System
-----
1. Add a new score board
2. Add score
3. Remove score
4. Search score
5. Display score board and score report
Your choice (1-6, other to quit):
```

- **Chức năng 1:** Thêm thông tin điểm của một môn học trong học kỳ. Các thông tin được nhập từ bàn phím bao gồm: Thông tin môn học, mã học kỳ, số lượng SV
- **Chức năng 2:** Thêm thông tin điểm môn học cho 1 sinh viên. Trước khi thêm cần hỏi người dùng mã môn học và mã học kỳ để thêm thông tin vào đúng file.
- **Chức năng 3:** Xóa thông tin điểm môn học cho 1 sinh viên. Cần hỏi người dùng mã môn học, mã học kỳ và mã sinh viên để xóa thông tin trên đúng file.
- **Chức năng 4:** Tìm kiếm thông tin điểm một môn học của sinh viên trong học kỳ nào đó. Người dùng cần cung cấp mã môn học, mã học kỳ, MSSV
- **Chức năng 5:** Hiển thị nội dung bảng điểm và báo cáo thống kê về điểm của môn học. Người dùng cần cung cấp mã môn học, mã học kỳ

Sau khi thực hiện xong mỗi chức năng, cần hỏi người dùng có tiếp tục sử dụng chức năng đó không. Nếu người dùng trả lời 'y' hoặc 'Y' thì thực hiện lại chức năng đó, ngược lại quay trở lại menu chính.

Tất cả các lớp của chương trình đặt vào gói **samsung.java.sms**

*Lưu ý: Ngoài các lớp và các thuộc tính, phương thức phải có như mô tả, nếu cần thiết sinh viên có thể viết thêm các lớp khác hoặc bổ sung thêm các thuộc tính, phương thức khác một cách tùy ý.*