

BLG458E - Functional Programming

Term Project

May 2020

Story

You are living in a Ninja World where there are 5 great countries and all of these countries are ruled by the ninjas ¹. These countries are Land of *Earth*, Land of *Lightning*, Land of *Water*, Land of *Wind* and Land of *Fire* ².

Each year, an exam called *Chunin Selection Exam* (CSE) is organized for the junior ninjas and this exam is an opportunity for them to be promoted to journeyman. This year, each county has decided to do their own examination as the first stage and select equal number of junior ninjas to send to CSE. They also decided to share the exam scores of the juniors with the CSE committee so that their examinations can also play a role for the decision of promotion to journeyman.

Since there are 5 countries and x number of selected ninjas, there will be $5*x$ juniors participated in the CSE. Each ninja gains several abilities during their training time and a junior ninja can obtain generally 2 abilities. In the CSE, ninjas will fight and the score of their fight will be decided according to their abilities. Each ability has a unique degree of importance and if a ninja has a greater ability compared to his/her opponent, he/she gets ahead of the game.

You are asked to write a program for the CSE so that some junior ninjas will be promoted to journeyman.

Details

Each county will send a report of their juniors' information to the CSE committee in the format of:

NameOfTheJunior NameOfTheCountry ExamScore1 ExamScore2 Ability1 Ability2
e.g. Sasuke Fire 50 60 Lightning Fire

At the end, the CSE committee will make a report of consisting $5*x$ lines. Each line will be about a 1 junior ninja and 1 ninja has 6 number of important features of his/her name, his/her country's name, his/her first exam score in the first stage that his/her country held, his/her second exam score in the first stage that his/her country held, his/her first ability and his/her second ability. All these features are given as 1 word and each 1 word of feature is separated with white space.

There are different kinds of abilities and a junior ninja can have 2 of them. These abilities and their impacts are listed as:

Clone: 20, Hit: 10, Lightning: 50, Vision: 30, Sand: 50, Fire: 40, Water: 30, Blade: 20,
Summon: 50, Storm: 10, Rock: 20

In the CSE, score of a junior in a fight is calculated as:

$$Score = 0.5 * ExamScore_1 + 0.3 * ExamScore_2 + Ability_1 + Ability_2$$

For example, in the *Sasuke Fire 50 60 Lightning Fire* case, Sasuke will have a score of 133 and he will pass the round if his opponent have a score less than 133 and fail if his opponent has a score greater than 133. If both of them have the same score, the one with the higher $Ability_1 + Ability_2$ score will pass. If again the scores are equal, 1 of them will pass the round randomly. 10 points will be added to the winner's score.

The report of the CSE committee will be given to your program as parameter. A sample report is shared as *csereport.txt* with $x = 3$. In your test cases, x can be different.

Implementation

In this section, UI of the program will be explained and some guidelines for the functions will be given.

You will name your module as **cse.hs** and you will use `ghc cse.hs -o cse` command to create an executable. Your program will be executed as `./cse csereport.txt`.

You can create a data type called **Ninja** as:

```
data Ninja = Ninja {name:: String, country:: Char,
                    status:: String, exam1:: Float,
                    exam2:: Float, ability1:: String,
                    ability2:: String, r:: Int}

-- name is the name of the ninja.
-- county is the county of the ninja. Character choice for each country is given below.
-- status is the current status of the ninja. It is initialized as junior.
-- exam1 is the score of the first examination.
-- exam2 is the score of the second examination.
-- ability1 is the first ability of the ninja.
-- ability2 is the second ability of the ninja.
-- r is the number of rounds that the ninja took place. It is initialized as 0.
```

Remember that this definition can change according to your design such as you may add *score* of a ninja to the definition.

You should have 5 different lists in your program such as:

```
fire :: [Ninja] -- add the junior ninjas of Land of Fire to that list
fire = []
lightning :: [Ninja] -- add the junior ninjas of Land of Lightning to that list
lightning = []
water :: [Ninja] -- add the junior ninjas of Land of Water to that list
water = []
wind :: [Ninja] -- add the junior ninjas of Land of Wind to that list
wind = []
earth :: [Ninja] -- add the junior ninjas of Land of Earth to that list
earth = []
```

These lists will be manipulated during the CSE and you will mainly use these lists. Since you will initialize them after you read the file which is given as a parameter, these definitions may also differ

according to your design. These are shown just to clarify that you will mainly deal with 5 different lists.

```
(base) gulcin@gulcin:~/Desktop$ ./cse csereport.txt
a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action:
```

UI of the program

a) View a Country's Ninja Information: (10 points)

With this option, the user will be asked to enter the county code. **e** or **E** is Land of Earth, **l** or **L** is Land of Lightning, **w** or **W** is Land of Water, **n** or **N** is Land of Wind and **f** or **F** is Land of Fire. When the user enters the country code, the program will show all of the ninjas of that country in the CSE, according to their scores in descending order. However, if the number of rounds for the ninjas of this country are not the same, the ones with the greater number of rounds will be listed later than the ones with the lower number of rounds even if their scores are greater than the previously listed ninjas.

Taking everything into consideration, the ninjas with the same number of rounds will be listed in descending order while the number of rounds will be in ascending order. Actually, the ordering of the ninjas in their country list will be also the same, meaning that making some rounds will affect the ordering of the ninjas in the country lists. Therefore, you will not need to reorder the ninjas in the country list for the purpose of viewing the ninjas. You will just iterate over the related country list.

Each country will promote only 1 ninja to journeyman. Therefore, if this country has already 1 promoted ninja, then a warning will be given to the user and any ninja from this country cannot be included to the fights anymore even if they are not disqualified. You can use a Boolean flag for each country in order to give this warning.

A ninja is promoted to journeyman if he/she wins 3 rounds. If none of the ninjas in a country does not satisfy this condition, the related country will not promote any ninja to journeyman.

```
a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: a
Enter the country code: f
Sasuke, Score: 133, Status: Junior, Round: 0
Naruto, Score: 103.5, Status: Junior, Round: 0
Neiji, Score: 82.5, Status: Junior, Round: 0
```

(a) Example for View a Country's Ninja Information

```
a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: a
Enter the country code: f
Naruto, Score: 123.5, Status: Junior, Round: 2
Sasuke, Score: 163, Status: Journeyman, Round: 3
Fire country cannot be included in a fight
```

(b) Example for the warning

b) View All Countries' Ninja Information: (15 points)

With this option, all of the ninjas that take the CSE will be listed according to their scores in descending order and their number of rounds in ascending order (the same rule described in *View a Country's Ninja Information*), meaning that all $5 \cdot x$ number of ninjas will be treated together. If some of the ninjas have the same score and same number of rounds, the ordering between them does not matter.

```

a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: b
Sasuke, Score: 133, Status: Junior, Round: 0
Gaara, Score: 131.5, Status: Junior, Round: 0
Sana, Score: 107, Status: Junior, Round: 0
Naruto, Score: 103.5, Status: Junior, Round: 0
Aimi, Score: 89.5, Status: Junior, Round: 0
Suiu, Score: 89, Status: Junior, Round: 0
Haruki, Score: 84.5, Status: Junior, Round: 0
Neiji, Score: 82.5, Status: Junior, Round: 0
Samidare, Score: 71.5, Status: Junior, Round: 0
Midare, Score: 71, Status: Junior, Round: 0
Miyazaki, Score: 69, Status: Junior, Round: 0
Temari, Score: 68, Status: Junior, Round: 0
Hiroshi, Score: 68, Status: Junior, Round: 0
Kira, Score: 68, Status: Junior, Round: 0
Kankuro, Score: 50, Status: Junior, Round: 0

```

Example for View All Countries' Ninja Information

c) Make a Round Between Ninjas: (25 points)

With this option, the user will be asked to enter the opponents' names and country codes. After the round is completed, the number of rounds for the winner will be incremented by 1 and his/her position in the country list will also change according to the rules described in *View a Country's Ninja Information*, in order to give a higher chance for the other ninjas of the same country to make a round by *Make a Round Between Countries* option (Details of that option will be explained later). If a ninja loses a round, he/she should be disqualified, meaning that he/she should be removed from the related country list. Example of a round between ninjas is shown and the effect of this round for the related country lists are also shown.

Any case of user mistake such as writing a name of ninja which is not in CSE or writing a wrong country code should be considered and meaningful error messages should be given to the user. Consider these kinds of cases for all options.

```

a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: c
Enter the name of the first ninja: Naruto
Enter the country code of the first ninja: f
Enter the name of the second ninja: Haruki
Enter the country code of the second ninja: e
Winner: "Naruto, Round: 1, Status: Junior"

```

(a) Example for Make a Round Between Ninjas

```

a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: a
Enter the country code: f
Sasuke, Score: 133, Status: Junior, Round: 0
Neiji, Score: 82.5, Status: Junior, Round: 0
Naruto, Score: 113.5, Status: Junior, Round: 1

```

(b) Land of Fire ninjas after a round

```

a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: a
Enter the country code: e
Miyazaki, Score: 69, Status: Junior, Round: 0
Hiroshi, Score: 68, Status: Junior, Round: 0

```

(c) Land of Earth ninjas after a round

d) Make a Round Between Countries: (20 points)

With this option, the user will be asked to enter 2 country codes. Then, the first ninjas of each country list will make a round. Since the ones who have the first place in their country lists are selected, that is why changing the order of the ninjas in the country lists after each round increases the chance for other ninjas to have a fight with this option. Again, after the winner is decided, the loser will be removed and the order of the winner will be changed in the related country lists.

```
a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: d
Enter the first country code: f
Enter the second country code: w
Winner: "Sasuke, Round: 1, Status: Junior"
```

(d) Example for Make a Round Between Countries

```
a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: a
Enter the country code: f
Neiji, Score: 82.5, Status: Junior, Round: 0
Sasuke, Score: 143, Status: Junior, Round: 1
Naruto, Score: 113.5, Status: Junior, Round: 1
```

(e) Land of Fire ninjas after a round

```
a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: a
Enter the country code: w
Samidare, Score: 71.5, Status: Junior, Round: 0
Midare, Score: 71, Status: Junior, Round: 0
```

(f) Land of Water ninjas after a round

e) Exit: (5 points)

With this option, the program will terminate after listing the ninjas who are promoted to journeyman. Normally, the program will always ask the user to enter an action (you may list the action options to the user every time you ask for an action, it is your choice) until this *Exit* option is selected. You can use country flags that are suggested in *View a Country's Ninja Information* section in order to list the journeymen.

```
a) View a County's Ninja Information
b) View All Countries' Ninja Information
c) Make a Round Between Ninjas
d) Make a Round Between Countries
e) Exit
Enter the action: e
Sasuke, Score: 163, Status: Journeyman, Round: 3
Gaara, Score: 161.5, Status: Journeyman, Round: 3
Sana, Score: 137, Status: Journeyman, Round: 3
```

Example for Exit and listing the journeymen

Evaluation

The distribution of 75 points is given with the explanations. The remaining 25 points will be about your program's efficiency, clarity and how much you use functional programming concepts in your implementation.

After the deadline, a schedule will be shared and we will hold Zoom meetings with each group as our demonstration sessions. Details of the meetings will be announced later.

You will use Github for your projects. You will create a private repository named as **BLG458E Group X** where *X* will be your group id. One of you can create this repository for your group and add others as collaborators. You can create a Github account with your ITU e-mail in order to create a private repository for free.

You should also add **baykalg@itu.edu.tr** as collaborator so that we can check your comments and work distribution. It will be also important for the grading.

Tips & Restrictions

- You need to use **System.IO** to handle I/O operations.
- You need to use **System.Environment** to handle command line arguments.
- Since you will manipulate the lists according to the given action, do not forget to work on the outputs of your functions (manipulated lists) when you want to perform new action.
- **This** page includes helpful examples for both I/O operations and string manipulations.
- You should use function compositions, function applications, higher order functions, list comprehensions and currying wherever they can be used. Therefore, your program should have at least **3** different kinds of higher order functions, at least **1** function composition and at least **1** currying.
- There will be a message board on Ninova and you will be able to ask for further restrictions on message board. It is your responsibility to check the message board regularly for further explanations.
- All the code you submit must be your own. You are not allowed to take any code from any resource.
- You are not allowed to use any construct that was not covered in the class.

You can ask your further questions about the homework via e-mail (baykalg@itu.edu.tr)