Lab 2

A. The Elo rating system is used to calculate the relative skill levels of players in competitor-versus-competitor games, such as chess. It is named after its creator Arpad Elo, a Hungarian-born American professor in physics.

If Player A has a rating of R_A and Player B a rating of R_B, the expected score for Player A is

$$E_A = \frac{1}{1+10^{(R_B-R_A)/400}}. \tag{1}$$

Similarly, the expected score for Player B is

$$E_B = \frac{1}{1+10^{(R_A-R_B)/400}}. \tag{2}$$

After a game is finished, if a player's actual score is larger than his expected score, the Elo system takes this as evidence that the player's rating is too low, and needs to be adjusted upward. Similarly, when a player's actual score is less than his expected score, the player's rating is adjusted downward.

Define K as the maximum possible adjustment per game. Suppose that Player A was expected to score E_A points but actually scored S_A points. The formula for updating his rating is

$$R'_A = R_A + K(S_A - E_A) \tag{3}$$

Define the actual score S_A as 1 for a win, as 0.5 for a draw, and 0 for a loss. Assume that K is 32, R_A is 1613 (for Player A), and R_B is 1609 (for Player B). From (1) and (2), we have $E_A = 0.506$ and $E_B = 0.494$. If Player A loses Player B, then rating R_A should be updated by (3) as

$$R'_A = 1613 + 32(0 - 0.506) = 1597,$$

and rating R_B should be updated as by (3) as

$$R'_B = 1609 + 32(1 - 0.494) = 1625.$$

Write a C++ program to calculate R_A and R_B for each game. Consider the following input file *file.in*:

```
32    1613     1609
0
0.5
0.5
0
0
```

0

The first line consists of K, initial R_A and initial R_B. From line two, each line represents the actual score (1 for win, 0.5 for draw, and 0 for loss) Player A gets at each game.

Your program reads in *file.in* and creates a file *file.out* with the following format:

```
1613  1609
1597  1625
1598  1624
1599  1623
1584  1638
1570  1652
1558  1664
```

For each line in *file.in* (starting from the second line), each line in *file.out* consists of the initial R_A and R_B values (the first line), and the calculated R_A and R_B values (starting from the second line). Note that R_A and R_B values are all rounded integers.

In your program, you need to design a class which records K, R_A and R_B values as private data members. To manipulate these data members, you need to provide corresponding get and set functions. You also need to provide a public member function for returning the calculated R_A (or R_B) value.

B. Execute the following command in your Unix platform:

$ **git clone https://github.com/ncku-pd2/pd2_sample.git**

to get the *Makefile* for sample code *timer6.cpp*, *timer6.h* and *main6.cpp* in lecture slides. Modify this file for your program in question A.

Hand-in Rules

Your GitLab account shall have a repository lab2, which includes the following things:

1. A *.cpp* and a *.h* files for each class.
2. A *.cpp* file for your main function.
3. A *file.in* file and a *file.out* file for testing. (You can copy the contents from question A or design it on your own.)
4. A *Makefile*.
5. A *README.md* file showing how to compile your program.