C PROGRAMMING - 20192

Description | Submission | Edit | Submission view

Lab Exam 1 Preliminary Question

Avaliable from: Thursday, 27 February 2020, 11:00 PM
Requested files: main.c (Download)
Type of work: Individual work
In this lab exam preliminary question, first, you will prompt the task number from the user. According to the task number, you will solve the following 2 tasks.

Task 1) Balance Calculator

In this question, you are asked to calculate the balance of a bank account after a couple of banking transactions given in a predefined format. Given an input starts with the initial balance of the account and continues with transactions. These transactions are not in Turkish Lira. They are in "E" Euro or "D" Dollar. You have to convert the transaction amounts (or volume of the transaction) into Turkish Lira according to the exchange rates provided in the input.

The input starts with the initial balance of the account which is a floating point number but **it can have a huge value**. It is followed by exchange rates of Euro and Dollar respectively in floating-point format. Then there comes the number of transactions given in the input and the rest of the input contains transactions.

There are two types of transactions: "D" deposit and "W" withdraw. Deposit means adding money to the account and withdraw means getting some money out of the account.

Each transaction is represented with two char and one floating-point input showing the type of the transaction, currency, and transaction amount respectively.

The rate of Euro to Turkish Lira means the equivalent amount of Turkish Liras for 1 Euro.

Valid input format is as below:
<task_number>
<initial_balance_in_TL> <rate_of_Euro_to_TL> <rate_of_Dollar_to_TL>

<number of transactions>

<type_of_transaction_1> <currency_1> <amount_1>

<type_of_transaction_2> <currency_2> <amount_2>

-
-
-

<type_of_transaction_n> <currency_n> <amount_n>

Valid output will be:

Final balance: <final_balance_in_TL> TL.

Specifications:

- The result will be a floating-point number with 2 digit precision.
- **There is a dot at the end of the output.**
- **The output does not contain a newline character.**
- Please obey the print format for getting points in the lab exam.

Sample input and outputs:

Input 1:

```
1
100.0 5.0 6.0
4
D D 10
W D 5
W E 2
D D 10
```

Output 1:

```
Final balance: 180.00 TL.
```

Input 2:

```
1
0.0 4.48 5.32
2
W E 5
W D 5
```

Output 2:

```
Final balance: -49.00 TL.
```

Input 3:

```
1
123456798.0 5.514 6.231
3
D D 123456.0
D E 654321.0
D D 111111.1
```

Output 3:

```
Final balance: 128526311.56 TL.
```

Input 4:

```
1
1111.1111 5.55 6.66
5
D D 11.11
W D 22.22
W E 33.33
D E 44.44
D D 55.55
```

Output 4:

```
Final balance: 1468.74 TL.
```

Task 2) Exam Evaluator

In this question, you are asked to evaluate university exam results given in a predefined format. Given an input starts with "T" (TYT) or "A" (AYT). The rest of the input varies according to the exam type. If the type of the exam is AYT then input continues with the subtype of the AYT exam. There are 3 different subtypes: "C" (Science-oriented), "O" (Social-oriented), and "E" (Equally-weighted).

- If the exam type is **TYT** there are no subtypes, there are 4 courses and each course has **40** questions.
- If the exam type is **AYT**, as mentioned above, there is a subtype. The number of courses changes depending on the subtype and each course has **30** questions.
    - If "C" or "O" then there are 2 courses.
    - If "E" then there are 4 courses.

After the exam type (and after subtype in case of AYT) inputs there comes a penalty score of wrong answers in floating-point format. We will see the calculation below.
The rest of the input contains 1 integer for the number of correct answers and 1 floating-point number for score per correct answer for each course.

The goal of the task is to calculate the total score of the student. The calculation of the score for a course is as follows:
(You can assume that there is no question that left empty in any exam.)

Total question: 40
Correct answers: 32
Score per correct answers: 2.0
Penalty: 0.2
(40 - 32 = 8 wrong answers)
>>
The Score of the course = 32 * 2.0 - (8 * 2.0)*0.2

Total question: 40
Correct answers: 15
Score per correct answers: 5.0
Penalty: 0.3
(40 - 15 = 25 wrong answers)
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
The Score of the course = 15 * 5.0 - (25 * 5.0)*0.3

Total question: 30
Correct answers: 17
Score per correct answers: 2.4
Penalty: 0.25
(30 - 17 = 13 wrong answers)
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
The Score of the course = 17 * 2.4 - (13 * 2.4)*0.25

- Exam type and subtype input are in **char** format.
- The numbers of correct answers in **integer** format.
- Penalty score of wrong answers and score per correct answer in **floating-point** format. **Do NOT use double variables.**

Valid input formats are as below:
<task_number>

T <penalty_score>

<#correct_1> <value_1> <#correct_2> <value_2> <#correct_3> <value_3> <#correct_4> <value_4>

Valid output will be:

T result: <total_point> points.

OR

<task_number>

A [C, O] <penalty_score>

<#correct_1> <value_1> <#correct_2> <value_2>

Valid output will be:

A-[C, O] result: <total_point> points.

OR

<task_number>

A E <penalty_score>

<#correct_1> <value_1> <#correct_2> <value_2> <#correct_3> <value_3> <#correct_4> <value_4>

Valid output will be:

A-E result: <total_point> points.

Specifications:

- The result will be a floating-point number with 2 digit precision.
- There are 40 questions in each course of TYT exam and there are 30 questions in each course of AYT exam.
- There is no question that left empty in any exam.
- **There is a dot at the end of the output.**
- **The output does not contain a newline character.**
- Please obey the print format for getting points in the lab exam.

Sample input and outputs:

Input 1:

```
2
T 0.1
5 1.1 5 2.2 5 3.3 5 4.4
```

Output 1:

```
T result: 16.50 points.
```

Input 2:

```
2
A C 0.15
8 3.5 12 5.2
```

Output 2:

```
A-C result: 64.81 points.
```

Input 3:

```
2
A O 0.5
0 5.0 0 5.0
```

Output 3:

```
A-O result: -150.00 points.
```

Input 4:

```
2
A E 0.4
0 3.0 10 2.0 20 1.0 30 0.5
```

Output 4:

```
A-E result: -1.00 points.
```

Requested files

main.c

```
#include <stdio.h>

int main ()
{
    int task;


    scanf("%d", &task);

    if(task == 1) {

        /* Solution for task 1.*/

    } else if(task == 2) {

        /* Solution for task 2.*/

    }

    return 0;
}
```

VPL