**Sabancı University**

Faculty of Engineering and Natural Sciences

**CS204 Advanced Programming**

**Fall 2018-2019**

**Homework 5 – Object Sharing**

**Due: 21/11/2018 - 23:55**

(One day late submission penalty: -10%)

|  |
| --- |
| **PLEASE NOTE:**  **Your program should be a robust one such that you have to consider all relevant programmer mistakes and extreme cases; you are expected to take actions accordingly!**  **You HAVE TO write down the code on your own.**  **You CANNOT HELP any friend while coding.**  **Plagiarism will not be tolerated!** |

1. **Introduction**

The aim of this homework is to practice object sharing concept in C++. In this homework you will implement three classes: account, parent, child. The parents and the child will share a *single* account and perform operations on it. The operations are explained in the document.

In this homework, there won’t be any algorithmic challenges. The aim is to practice object sharing concept, namely sharing an account object between different classes. The source file is given **and you are not allowed to change it.** You are only going to implement the related classes.

1. **Program Flow**

There will be three classes to implement.

* + - **Account**: This class will implement a basic account object. There will be three member variables and they will be *double* values. The members are: *balance, limit, debt.* You need to implement appropriate functions that will make the source file work.
    - **Parent:** This class will implement a basic parent object. There will be two member variables: *name* and *account.* The parent can perform the following functions:
      * **printAccountInfo() :** This function will display the current information about the account. For the exact output, please check Sample Runs.

Note: The available account limit is calculated as ***limit – debt*.**

* + - * **depositMoney(double) :** This function will deposit the given amount to the account. After the deposit, the balance will be increased by the given amount.
      * **spendMoney(double) :** This function will try to spend the given amount of money. If the ***balance*** is more than the amount to spent, only ***balance*** will decrease. If there are not enough money in ***balance***, the remaining money will be added to ***debt***. If the debt exceeds the ***limit,*** an error message will be prompted and no money will be spent.
      * **payDebt(double) :** This function will try to pay the given amount of debt. If the given amount is more than the current debt, an error message will be prompted. Otherwise, the debt will be paid.
    - **Child:** This class will implement a basic child object. The child can only display the account info (the balance and the available limit) and spend money. While spending money, a child can only spend up to 200TL. Otherwise, an error message will be displayed.

1. **Sample Runs**

The given source file produces the following output. Please be careful about the constructors and **DO NOT CHANGE THE SOURCE CODE.**

Lily has deposited 50TL.

Lily has successfully spent 250TL.

James has successfully spent 2000TL.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

The account balance is: 0TL.

The account debt is: 1700TL.

The account limit is: 2000TL.

The available account limit is: 300TL.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Harry can not spend 500TL. Children can spend up to 200TL.

Harry has successfully spent 50TL.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

The account balance is: 0TL.

The available account limit is: 250TL.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

James attempts to pay more than debt. No payment is done.

James payed 1000TL of debt.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

The account balance is: 0TL.

The account debt is: 750TL.

The account limit is: 2000TL.

The available account limit is: 1250TL.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Lily payed 750TL of debt.

James has deposited 500TL.

Harry can not spend 250TL. Children can spend up to 200TL.

Harry has successfully spent 150TL.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

The account balance is: 350TL.

The account debt is: 0TL.

The account limit is: 2000TL.

The available account limit is: 2000TL.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Press any key to continue . . .

**Some Important Rules:**

In order to get a full credit, your programs must be efficient and well presented, presence of any redundant computation or bad indentation, or missing, irrelevant comments are going to decrease your grades. You also have to use understandable identifier names, informative introduction and prompts. Modularity is also important; you have to use functions wherever needed and appropriate.

When we grade your homeworks we pay attention to these issues. Moreover, in order to observe the real performance of your codes, we may run your programs in *Release* mode and **we may test your programs with very large test cases**.

**What and where to submit (PLEASE READ, IMPORTANT):** You should prepare (or at least test) your program using MS Visual Studio 2012 C++. We will use the standard C++ compiler and libraries of the abovementioned platform while testing your homework. It'd be a good idea to write your name and last name in the program (as a comment line of course).

Submissions guidelines are below. Some parts of the grading process are automatic. Students are expected to strictly follow these guidelines in order to have a smooth grading process. If you do not follow these guidelines, depending on the severity of the problem created during the grading process, 5 or more penalty points are to be deducted from the grade.

Name your cpp file that contains your program as follows:

***“SUCourseUserName\_YourLastname\_YourName\_HWnumber.cpp”***

Your SUCourse user name is actually your SUNet username that is used for checking sabanciuniv e-mails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name. For example, if your SUCourse user name is cago, name is Çağlayan, and last name is Özbugsızkodyazaroğlu, then the file name must be:

***Cago\_Ozbugsizkodyazaroglu\_Caglayan\_hw2.cpp***

Do not add any other character or phrase to the file name. Make sure that this file is the latest version of your homework program. Compress this cpp file using WINZIP or WINRAR programs. Please use "zip" compression. "rar" or another compression mechanism is NOT allowed. Our homework processing system works only with zip files. Therefore, make sure that the resulting compressed file has a zip extension. Check that your compressed file opens up correctly and it contains your cpp file.

You will receive no credits if your compressed zip file does not expand or it does not contain the correct file. The naming convention of the zip file is the same as the cpp file (except the extension of the file of course). The name of the zip file should be as follows:

***SUCourseUserName\_YourLastname\_YourName\_HWnumber.zip***

For example zubzipler\_Zipleroglu\_Zubeyir\_hw1.zip is a valid name, but

***Hw2\_hoz\_HasanOz.zip, HasanOzHoz.zip***

are **NOT** valid names.

**Submit via SUCourse ONLY!** You will receive no credits if you submit by other means (e-mail, paper, etc.).

Successful submission is one of the requirements of the homework. If, for some reason, you cannot successfully submit your homework and we cannot grade it, your grade will be 0.

Good Luck!

CS204 Team (Mustafa Kemal Taş, Kamer Kaya)