Assignment 3

Seon Joo Kim
Yonsei University

Outline

- Deadline
- No Plagiarism
- Leave Comments
- Scoring
- Problem 1
- Problem 2
- Submission
- Questions

Deadline

- Tuesday, nov 17th 23:55
- No late submissions at all

No Plagiarism

- No Mercy.
- The punishment will be made to both
  - the person who copied the code, and the person who shared the code.
- We will do plagiarism test with codes that were made in previous semesters and also in google. So be careful ☺

Leave Comments

- Leave comments in your file for TAs to understand your code.
- If no comments in the file, there may be a reduction of points.

Scoring

- You should take care of your code not terminating by an issue in the middle of the loop
  - Scores will be given only by the final outputted file

- **Problem1 (40pt)**
  - 4 test cases
  - If your code is correct as O X O O O if ran separately but terminates in the second test case by an error only the first test case is considered correct
  - If your code output is correct for given example input1.txt file, we will give you base score 10pt.

- **Problem2 (60pt)**
  - If your code output is correct for given example in this pdf, we will give you base score 20pt.
  - In problem2, We do not provide skeleton code.
  - Since the input is received by cin, the user have to enter the input in terminal directly.

Scoring

- When ran the code, the printed text *(by cout)* in the terminal can be recorded in a file by '>>' command
- We will score the results by saving your programs printed texts by '>>', and compare by 'diff' command
- problem1, problem2 will be graded in this way.
- Example for problem 1

```
$ g++ -Wall problem1.cpp –o problem1
$ ./problem1 >> output1.txt
$ diff answer.txt output1.txt
```

Problem 1

- complete a *Matrix* class by overloading basic operator + , - , *
  - If m1, m2 are *Matrix* object, and m1 = $\begin{bmatrix}1 & 0\\2 & 2\end{bmatrix}$, m2 = $\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}$
  - m1+m2 = $\begin{bmatrix}2 & 2\\5 & 6\end{bmatrix}$, m1-m2 = $\begin{bmatrix}0 & -2\\-1 & -2\end{bmatrix}$, m1*m2 = $\begin{bmatrix}1 & 2\\8 & 12\end{bmatrix}$ (actual result of operator +,-,* is *Matrix* object)
  - In this program, * is matrix multiplication ! If you don't know about matrix multiplication, read this link
  - cout<<m1*m2; -> this code should print
```
1 2
8 12
```
- Also, you have to make transpose function which returns transpose matrix of object matrix.
  - m1.transpose() = $\begin{bmatrix}1 & 2\\0 & 2\end{bmatrix}$, m2.transpose() = $\begin{bmatrix}1 & 3\\2 & 2\end{bmatrix}$
  - This is const function. Do not modify object value. just generate another Matrix object that contains transpose matrix, and return it

Problem 1

- Instruction
  - Input
    - First row of input is size of matrix : N
    - **Assume inputs are two N X N matrices. (1≤N<100)**
    - From the second row to NxN+1 row, each line contains value of matrix 1.
    - From the NxN+2 to 2NxN+1 row, each line contains value of matrix 2.
    - Don’t touch anything without where you supposed to write your code.
    - No error case for this input, so don’t need to handle exception for this input.
    - Only integer is allowed for input. (Do not care about exception input)

Problem 1

- Two I/O example

Input (input1.txt)	Output
2 1 0 2 2 1 2 3 4	matrix 1: 1 0 2 2 matrix 2: 1 2 3 4 sum of matrix 1 and matrix 2: 2 2 5 6 subtract matrix 2 from matrix 1: 0 -2 -1 -2 transposed matrix 1: 1 2 0 2 transposed matrix 2: 1 3 2 4 multiplication of matrices: 1 2 8 12

2x2 matrix

matrix 1

matrix 2

Input (input1.txt)	Output
3 1 0 3 0 2 1 6 8 2 4 9 0 1 5 2 2 0 4	matrix 1: 1 0 3 0 2 1 6 8 2 matrix 2: 4 9 0 1 5 2 2 0 4 sum of matrix 1 and matrix 2: 5 9 3 1 7 3 8 8 6 subtract matrix 2 from matrix 1: -3 -9 3 -1 -3 -1 4 8 -2 transposed matrix 1: 1 0 6 0 2 8 3 1 2 transposed matrix 2: 4 1 2 9 5 0 0 2 4 multiplication of matrices: 10 9 12 4 10 8 36 94 24

3x3 matrix

matrix 1

matrix 2

Tip – Method of menu implementation

```
char option;
while(true)
{
    cin >> option;

    switch(option)
    {
        case ‘1’:
          //menu 1
          break;
        case ‘2’:
          // menu 2
          break;
      . . . // You can make the program with menu like this.
        case ‘0’:
          // exit example
          return 0;
        default :
          // Print the error message, and return back to menu
          continue;
    }
}
```

Problem 2
Mcdonalds Delivery Systems

Goal : Exercise of class implementation

Program Overview

- This project is an implementation of Mcdonalds Delivery System.
- The number of delivery product is 8.
- Each user has an id, password, initial charge amount, order history. (id should be unique)
- If you want to order some product, you choose some product in the order menu.
- The program has the ability to refund orders.
- **You should make the program based on Class.**
- **The rules of Class name, member function, # of member function are free.**
- **But you should encapsulate data (hide personal data ex. id, password, money, order history)**
- **You can use multiple classes.**
- **In problem2, you should get input by cin>>**
- **Don't care about invalid type of input.**
- **Please, make the program exactly same as shown in the following slide!!**

Problem 2

<Main Menu>

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select ->
```

Problem 2

- **Program Details (Total score: 60pts)**

* Make the member function or function for the following problem.

1. Sign up(10 pts)

- **Implement the 'sign up' section. Input is ID, password, initial charge amounts.(Max # of account: 100 / Max length of Id, PW: 10 / If entered ID exists, then print the error message and Enter the id again.**
- **If you already logged in, then print the error message and back to main**
- **Don't care about exceed Max # of account, Max length of ID, PW.**

Problem 2

1. Sign up(10pt)

```
----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 1

Input your ID(0-to main menu) : OOP
Input your password : 1234
Input your total money : 10000

Welcome, OOP

----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 1

Input your ID(0-to main menu) : OOP2
Input your password : 1232
Input your total money : 20000

Welcome, OOP2
```

Highlight text is your input

You have to get input using cin<<

← Make the account
5pt

← This program can have multiple accounts(Max number: 100)

Problem 2

1. Sign up(10pt)

```
----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 1

Input your ID(0-to main menu) : OOP
Input your password : 1234
Input your total money : 10000

Welcome, OOP

----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 1

Input your ID(0-to main menu) : OOP
Duplicated user id

Input your ID(0-to main menu) :
```

Highlight text is your input

You have to get input using cin<<

you don't have to care about invalid type of input.

ex) assume always enter string for id/pw, integer for total money

Exception handling
5pt

← if user create the account with an existing ID, Print the error message and enter the other ID again.

Problem 2

1. Sign up(10pt)

```
----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 2

Input your ID(0-to main menu) : OOP
Input your password : 1234

Welcome, OOP

----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 1

You are already logged in to OOP account
Return to main menu

----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------
```

Highlight text is your input

You have to get input using cin<<

← login process(details in 2.)

← if you already logged in, print error msg and return to main menu

Problem 2

- **Program Details (Total score: 60pts)**

* Make the member function or function for the following problem.

2. Login(10 pts)

- **Implement the 'login' section. Input is ID, password.**

5pt

- **If you already logged in, then print the error message and back to main**
- **If entered ID doesn't exist, then print the error message and back to main menu.**

5pt

- **If entered ID exists, then enter pw.**
- **if entered pw doesn't match, then print the error message and ask pw again**
- **if entered pw matches, then print the welcome message and login to account.**

Problem 2

2. login(10pt)

```
---------------------------------
-----Mcdonalds Delivery System-----
---------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 2

Input your ID(0-to main menu) : OOp
ID is not existing
---------------------------------
-----Mcdonalds Delivery System-----
---------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 2

Input your ID(0-to main menu) : OOP
Input your password : 12
Wrong password!
Input your password : 1234

Welcome, OOP
```

Highlight text is your input

You have to get input using cin<<

← if entered id is not exists, print error message and go to main menu

← if entered pw is not match with id account, print error message and ask pw again.

Problem 2

2. login(10pt)

```
----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 2

Input your ID(0-to main menu) : OOP
Input your password : 12
Wrong password!
Input your password : 1234

Welcome, OOP

----------------------------------
-----Mcdonalds Delivery System-----
----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 2

You are already logged in to OOP account
Return to main menu

----------------------------------
-----Mcdonalds Delivery System-----
```

Highlight text is your input

You have to get input using cin<<

← if entered pw is corect, print welcome message and go to main menu

← if you already login, and didn't logout yet, print error message and back to main menu

Problem 2

- **Program Details (Total score: 60pts)**

* Make the member function or function for the following problem.

3. Show menu(5 pts)

- **The section of showing menu. Please refer to the following slide for the information of products.**

Problem 2

3. Show Menu(5pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 3
```

Highlight text is your input

You have to get input using cin<<

```
----------Mcdonalds Menu----------
1.  Hamburger               - 1000
2.  Cheeseburger            - 1500
3.  Big Mac                 - 3000
4.  Quarter Pounder Burger  - 4000
5.  Double Quarter Pounder  - 5000
6.  Fries                   - 1500
7.  Chicken McNuggets       - 2000
8.  Coke                    - 1500
```

← print all foods of mcdonalds

You can print menu regardless of whether you are logged in or not.

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
```

← after than, back to main menu

Problem 2

- **Program Details (Total score: 60pts)**

* Make the member function or function for the following problem.

4. Order(10 pts)

- **The section of ordering the product after login process.**
- **If you are not logged in, print error message and return to main menu.**
- **Ask the user which food to buy and the amount to purchase, until the user input reaches 0.**
- **If the user input is 0, order is complete and print the total order.**

*** If the total order amount exceeds the balance, print the error message**

Problem 2

4. order(10pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 4

You have to login first
Return to main menu

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------
```

Highlight text is your input

You have to get input using cin<<

← If you didn't login before, print error message and return to main menu

Problem 2

4. order(10pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 4

Select menu number(0-to main menu) : 2
How many 'Cheeseburger' do you want : 1

Select menu number(0-to main menu) : 3
How many 'Big Mac' do you want : 1

Select menu number(0-to main menu) : 8
How many 'Coke' do you want : 2

Select menu number(0-to main menu) : 0
Your total purchase amount is 7500
Your balance is 2500
Thanks

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------
```

Highlight text is your input

You have to get input using cin<<

How many '~~' do you want

('~~' is food name of the selected number)

← select the menu and # of food

← If input is 0, the current order state is printed

← after than, back to main menu

Problem 2

4. order (10pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 4

Select menu number(0-to main menu) : 3
How many 'Big Mac' do you want : 1

You don't have enough money!

Select menu number(0-to main menu) : 6
How many 'Fries' do you want : 1

Select menu number(0-to main menu) : 5
How many 'Double Quarter Pounder' do you want : 0

Select menu number(0-to main menu) : 0
Your total purchase amount is 9000
Your balance is 1000
Thanks

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------
```

Highlight text is your input

You have to get input using cin<<

When your balance is 2500, you can't buy Big Mac. Over the budget.

Print error msg, and ask again

← If you enter a purchase quantity of 0, it is considered not purchased.

← after than, back to main menu

Problem 2

- **Program Details (Total score: 60pts)**

* Make the member function or function for the following problem.

5. Order condition (5pts)

- **The section of checking current user order history.**
- **If you are not logged in, print error message and return to main menu.**
- **The sequence of order history is same as the order time.**
- **Print the ordered food and amounts.(5pts)**
- **Do not print about the food item you have not ordered.(-2pts)**

+ Print the total purchase amount and balance.

Problem 2

5. Order condition(5pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 5

You have to login first
Return to main menu

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------
```

Highlight text is your input

You have to get input using cin<<

← If you didn't login before, print error message and return to main menu

Problem 2

5. Order condition(5pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 5

1. 'Cheeseburger'
2. 'Big Mac'
3. 'Coke'
4. 'Coke'
5. 'Fries'

Your total price is 9000
Your total money is 1000

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------
```

Highlight text is your input

You have to get input using cin<<

← Print the current **order history**(keep the sequence of order) and **total purchase amount and balance.**

← back to main menu

Problem 2

- **Program Details (Total score: 60pts)**

* Make the member function or function for the following problem.

6. refund(10pts)

- **Refund section for ordering food.**
- **If you are not logged in, print error message and return to main menu.**
- If the user selects the item to be refunded, item will be refunded. You have to get refund money back **(5pts)**
- **Print the error message for wrong input and try typing again.(5 pts)**

Problem 2

6. Refund(10pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 6

You have to login first
Return to main menu

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------
```

Highlight text is your input

You have to get input using cin<<

← If you didn't login before, print error message and return to main menu

Problem 2

6. Refund(10pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 6

1. 'Cheeseburger'
2. 'Big Mac'
3. 'Coke'
4. 'Coke'
5. 'Fries'
Your total price is 9000
Your total money is 1000

Choose number(0 - to main menu) : 2
Successfully returned

1. 'Cheeseburger'
2. 'Coke'
3. 'Coke'
4. 'Fries'
Your total price is 6000
Your total money is 4000

Choose number(0 - to main menu) : 3
Successfully returned

1. 'Cheeseburger'
2. 'Coke'
3. 'Fries'
Your total price is 4500
Your total money is 5500
```

Highlight text is your input

You have to get input using cin<<

← Print the current **order condition**(keep the sequence of order) and ask what to refund, select refund item

← Print the current **order condition** and ask what to refund again, after item refunded

Problem 2

6. Refund(10pt)

```
1. 'Cheeseburger'
2. 'Coke'
Your total price is 3000
Your total money is 7000

Choose number(0 - to main menu) : 3
Wrong input!

Choose number(0 - to main menu) : 2
Successfully returned

1. 'Cheeseburger'
Your total price is 1500
Your total money is 8500

Choose number(0 - to main menu) : 1
Successfully returned

Your total price is 0
Your total money is 10000

Choose number(0 - to main menu) : 1
Wrong input!

Choose number(0 - to main menu) :
```

Highlight text is your input

You have to get input using cin<<

← if you enter a number that is not in the list, print error msg and ask again

← when order food list is empty, print total price, total balance(money). Ask again until user enter 0

Problem 2

- **Program Details (Total score: 60pts)**

* Make the member function or function for the following problem.

7. logout(10pts)

- **logout section**
- **If you are not logged in, print error message and return to main menu.**
- If the user enter 0, logout process should be cancelled**.**
- If the user enter 1, logout from the current account.
- **Print the error message for wrong input(all input except 0, 1) and ask again.**

Problem 2

7. logout(10pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 7

If you want to logout, Please enter 1 (0-to main menu) : 3
Wrong input!
If you want to logout, Please enter 1 (0-to main menu) : 0
Logout is canceled

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------
```

Highlight text is your input

You have to get input using cin<<

5pt) if you enter wrong input, print error msg and ask again.

If 0 is entered, cancel logout process

← return to main menu

Problem 2

7. logout(10pt)

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 7

If you want to logout, Please enter 1 (0-to main menu) : 1

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 7

You have to login first
Return to main menu

-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------
```

Highlight text is your input

You have to get input using cin<<

5pt) If 1 is entered, logout from current account

← if you are not logged in, print error msg and return to main menu

Problem 2

8. Exit

```
-----------------------------------
-----Mcdonalds Delivery System-----
-----------------------------------

1. Sign up
2. login
3. Show Menu
4. Order
5. Order condition
6. Refund
7. Logout
0. Exit
Select -> 0
End the program
```

Highlight text is your input

You have to get input using cin<<

If the user input is 0 in the main menu, exit the program

Submission

- Zip the folder by running following command correctly
  - tar –zcvf 2020123456_hw3.tar.gz 2020123456_hw3/

- studentId_hw3.tar.gz
  - Ex) 2020123456_hw3.tar.gz
- There is going to be reduction of points if not following the folder hierarchy as well
- If unzipped your submission .tar.gz file should follow the folder hierarchy below

Current directory

- studentId_hw3.tar.gz
- studentId_hw3
  - problem1.cpp
  - problem2.cpp

Questions

- Use oop20202@gmail.com for questions

- We are not going to answer
  - Questions sent to TAs' personal mails
  - Questions not making sense
  - Questions related to the algorithm for solving the question
  - Questions you can infer the answer if read this file thoroughly
  - Questions you can simply solve by googling
    - Ex) how do I make a folder on ubuntu?

Appendix

- Zipping and unzipping the folder by tar command
  - https://linuxize.com/post/how-to-extract-unzip-tar-gz-file/
  - https://www.cyberciti.biz/faq/how-do-i-compress-a-whole-linux-or-unix-directory/