
Exercise 1 (Re-Take)

Class Constructors and Destructor

Duration : 75 minutes (including submission)

Exercise Materials

- Program templates are provided for this exercise. Please download from the elearning and extract the ZIP file to your local drive.
- The questions and tasks to be completed are stated in the program templates.
- You have the choice to use Microsoft VS Code, Dev C++ or any other IDEs to write the code for this exercise.

Deliverable Item

- Submit only the **source code** file you have edited (i.e., **exercise1.cpp**)
- The submission must be done at elearning. Other than that, e.g., email, telegram, etc. is not acceptable.

Plagiarism Policy

- Discussions among the students are still possible during the exercise session.
- However, all works must be done individually.
- Any kind of plagiarism (e.g., copy and paste code by any mean) would lead to disqualification of submissions for both parties (i.e., students that copy others' code and students that give their code to others).

Late Submission Policy

- **10% deduction for every 5 minutes late.**
- For example: if the duration of the exercise is 75 minutes, and your submission is received on the 76th minute, you are only eligible to earn 90% at maximum of the total marks.

Question

In this exercise, you will be defining a class named `Date` consisting of the attributes, `day`, `month` and `year`. Complete the program **exercise1.cpp** according to the following tasks:

Task 1:

Declare and define the class `Date`. The attributes have already been declared in the class declaration. Also, a method named `print` has also been given in the class.

Add the following methods:

- A default constructor.
- A constructor that accepts three parameters (i.e. `day`, `month` and `year`), respectively. The year must be stored as a four-digit value (e.g., 2018, 1995). However, this constructor might receive the parameter as a two-digit value (e.g. 18 or 95). The constructor should be able to handle such cases.
- A destructor which shows a message, for example **“The destructor is being executed”**.
- Accessor (or getter) methods for all the attributes, i.e. `getDay()`, `getMonth()`, and `getYear()`
- Mutator (or setter) methods for all the attributes. Special attention has to be given to the mutator `setYear()`. It might accept the parameter of a two-digit or four-digit year value. However, the class must store the year as a four-digit value. The following shows the conversion that this method needs to do:

If the **parameter <19**, the year is set to **20XX**,

If the **parameter <100**, the year is set to **19XX**

Other than that, simply consider it as a four-digit.

Task 2:

- In the main function, create a `Date` object for today's date (e.g, 9-9-2018)
- and print the date onto the screen using an appropriate method.

Task 3:

- Create another `Date` object for tomorrow's date. However, you must send the year parameter as 18 (instead of 2018).
- and print the date.

Task 4:

- a. Another `Date` object named `anotherDay` has been created in the main function. Also, the code to read user inputs from the keyboard has also been given. Complete this task such that the object attributes are set with these user inputs.
- b. and print the date onto the screen.

Output:

Expected result from the program is as shown in the following figure. **Bold** text indicates keyboard inputs.

Run 1

```
The date is 9-9-2018  
The date is 10-9-2018  
Enter the the date (day month year) => 30 12 2018  
The date is 30-12-2018
```

Run 2

```
The date is 9-9-2018  
The date is 10-9-2018  
Enter the the date (day month year) => 1 1 10  
The date is 1-1-2010
```

Run 3

```
The date is 9-9-2018  
The date is 10-9-2018  
Enter the the date (day month year) => 9 9 99  
The date is 9-9-1999
```