**Important Note:**

* **Define your class in “Date.h” file, Implement all the functions of Date class in “Date.cpp” and Test your class in “Driver.cpp”**

**Exercise 1:**

* Create a class **Date** having following private data members:

Int Day

Int Month

Int Year

* Create an object of Date “***date1***” and run your program

**Exercise 2 [Default Constructor]:**

* Write a default Constructor of Date that initializes the object to 1st January 1926 and prints “*Default Constructor Called*” in start
* Now run your program and test what does date1 prints?

**Exercise 3 [Print Function]:**

* Implement a function **Print** in Date class which prints a date in following format:

dd/mm/yyyy (e.g. 1/1/1926 for date1)

* Print object date1 in your main function and run the program.
* What does it print and how can we initialize the data of date1 at the time of creation?

**Exercise 4 [Overloaded Constructor with Default Argument]:**

* Write an overloaded Constructor of Date class that initializes the date object to date, month and year provided as parameter and prints “Overloaded Function Called”
* Now create another object ***independanceDay*** in main that is 14/08/2018
* Print ***independanceDay*** by calling Print function of Date class and run your program

**Exercise 5 [Destructor]:**

* Write Destructor of Date class that prints “Destructor called”
* Run your program and test it

**Exercise 6 [Input Function]:**

* Write a function **Input** in your Date class that takes input from user to populate a Date object
* Call “*date1.Input()*” and “*date1.Print()*” in your driver program and test it

**Exercise 7 [Setters]:**

* Create an object ***xmasDay*** using default constructor
* Print xmasDay and see what it prints
* Write Setters i.e. SetDay, SetMonth and SetYear in your class
* Now set xmasDay to 25/12/2018 using Setters in main

**Exercise 8 [Getters]:**

* Write Getters i.e. GetDay, GetMonth and GetYear in your date class
* Now print xmasDay using Getters in your Driver program

**Exercise 9 [Built-in Assignment Operator]:**

* Create an object **temp** of Date class
* Assign value of xmasDay to temp
* Print temp and test your program

**Exercise 10 [Passing object by value]:**

* Write a function **int Compare(Date)** that compares two dates, returns 1 if left hand side object is greater than right hand side object, -1 if lhs is smaller and 0 otherwise
* Test your function

**Exercise 11 [Return object by value]:**

* Write a function **Date IncrementMonth()** that returns a newly created Date object with one month next to the current date object. For example, if date1 = 2/01/2016 *date1.IncrementDate()* will return 2/02/2016 without changing date1
* Print both the date1 and newly created date in your driver program to test the result