**National University of Computer and Emerging Sciences**



**Lab Manual 03**

**Object Oriented Programming**

|  |  |
| --- | --- |
| Course Instructor | Ms. Arooj Khalil |
| Lab Instructor (s) | Ms. Fariha Maqbool  Ms. Amara Nasir |
| Section | BSE-2A |
| Semester | Spring 2023 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

## Objectives

After performing this lab, students shall be able to:

* Learn basics of classes and object creation.
* Create Constructors (Default Parameterized, Overloaded).
* Create and use setters and getters.

**TASK 1:**

Define your class in “Student’s” file, Implement all the functions of Student class in “Student.cpp” and Test your class in “Driver.cpp”.

**Exercise 1:**

* Create a class Student having following private data members:
* string roll-number
* Int age
* Float cgpa
* Create a public member function **print** in Student class which prints a Student info in following format: roll#/age/cgpa (e.g. 21L-1234/ 25/ 3.24 for std\_1)
* Create an object of Student “std\_1” and call the function print. run your program

**Exercise 2 [Default Constructor]:**

* Write a default Constructor of Student that initializes the objects with the default values (21L-1234, 25, 3.24) and prints a message “Default Constructor Called”.
* Call the print function and run your program and test what does std\_1 print?

**Exercise 3 [Overloaded/Parameterized Constructor]:**

* Write an overloaded/parameterized Constructor of Student class that initializes the data members of student class with the received parameters (r , a and c) and prints “Overloaded/parameterized Constructor Called”
* Now create another object std\_ 2 in main and pass the data in parameters as ("21L-1234", 19, 3.46);
* Print Student 2 by calling Print function of student class and run your program

**Exercise 4 [getInput Function]:**

* Create a public function getInput in your student class that takes input from user to populate a Student object.
* Call “std\_1.getInput()” and “std\_1.print()” in main function and test it.

**Exercise 5 [Setters/getters]:**

* Create separate setter functions for each data member and similarly constant getter functions for each data member.
* Dynamically create an object std\_3 using default constructor.
* Call the setter functions for std\_3 and pass the relevant data into the parameters
* call the getter functions from main function for std\_3 and display the data inside main function

**Task#2:**

* Create a class Student with data members (size, an integer variable and marks, a pointer to integer and it should be used to declare an array inside the constructor).
* create a Default constructor that should initialize the pointer with NULL and size with 0, and a parameterized constructors that should receive an integer array and size as parameter.

**Note :** Parameterized constructors should receive an array from main function and declare and initialize the array ‘marks’ with the received array i.e., (implement deep copy)

* Now Create a function “scale up”. This function should increase the marks of each student by 5.

**Note:** make sure that by adding the value 5 the updated value should not exceed 100 and similarly don't apply this scale up factor on the marks less than 45.

* Create a function “printInfo” to print the data.
* create a static array Inside main function and initialize with hard coded values. You can also initialize with user input.
* Now create the object of class and pass the local array, and its size into the parameters.
* Call printInfo function and then Call the function “scale up”.
* Call the printInfo function again after calling the scale up function and check whether the data is updated or not and whether it is in the valid range or not.
* Create a destructor inside the class that should de-allocate the dynamically allocated memory and initialize the pointer with nullptr to handle dangling pointer.

**TASK#3:**

As we already know that a class is simply a representation of a type of object. It is the blueprint/ plan/template that describes the details of an object.

Your task is to design a class for **Student.** You must take any 5 necessary data members as **Roll No, Name, CNIC, Degree and Address**. implement all the required functions i.e. default constructors, parameterized constructor, getters, setters, getInput function, print function.

Now create a static array of objects of **3 to 4 size** like Student stdArr[3] and test all the functions of the class.

**NOTE: Make sure that you create 3 separate files, .h file to declare class and its members, imp.cpp file for implementation member function of class. Driver.cpp file for main() function.**