Air University

Mid Semester Examinations: Spring 2025

Student ID: _____

Subject: OOP Lab
Class: BYCYS - F24ev
Section(s): A, B
Course Code: CS112L

Time Allowed: 60 Minutes
Max Marks: 50
FM's Name: Iram Fatima Hashmi
FM's Signature:

INSTRUCTIONS

- All questions must be attempted.
- This examination carries 15% weight towards the final grade.
- Submit code in word file in text form (no screenshots). And submit output screenshot of entire screen as shared in lab in same file.

Q. No. 1 (CLO 1)	**A university needs a mechanism to manage student records. Implement a Student class and perform following.** **a)** Implement a Student class having following data members: name, id, department (pointer type), cgpa, totalStudents. **b)** Create a constructor with initializer list to the class Student to initialize the data members of object. Also, create a copy constructor to create copy of objects. **c)** Create getter and setter functions. Also, create a member function changeDep() that changes the department of student and at the end displays data of student. **d)** Use your understanding of static and constant keyword to make the members constant and static. **e)** In main function perform the following: i. Create an array of 5 objects (2 default and 3 parameterized). ii. Create a pointer type object. iii. Create an object that is copy of object created in part ii. Mention if it's shallow copy or deep copy. iv. Call changeDep() function to change department of object created in part ii.	**Marks (30)** 5+5+5+5+10
Q. No. 2 (CLO 1)	**Below is the code for Employee class having one data member id. Main function is also provided.** class Employee { int* id;	**Marks (20)** 15+5

```
public:
        Employee(int idEmp) : id(new int(10)){
        }
        Employee(const Employee& emp) {

        id = emp.id;
        }
        void getID() const {

        cout << "ID= " << *id << endl;
        }
        void setID(int empID) {

        *id = empID;
        }
};
int main() {
        Employee* emp = new Employee(2);
        Employee* emp2 = emp;
        emp->setID(3);
        emp->getID();
        emp2->getID();
}
```

a) In main function “emp2” is created as copy of “emp”. But upon updating “id” of “emp”, “id” of “emp2” also gets updated. Identify the issue and fix it so that when “id” of “emp” is changed there’s no effect on “id” of “emp2”. Also identify any other logical issues if they exist and fix them.

b) Create a flow showing how the code was working before fixing the issue and after fixing the issue.