

Topics covered in this lab:

1. Aggregation
2. Composition
3. Polymorphism and virtual function

Section A

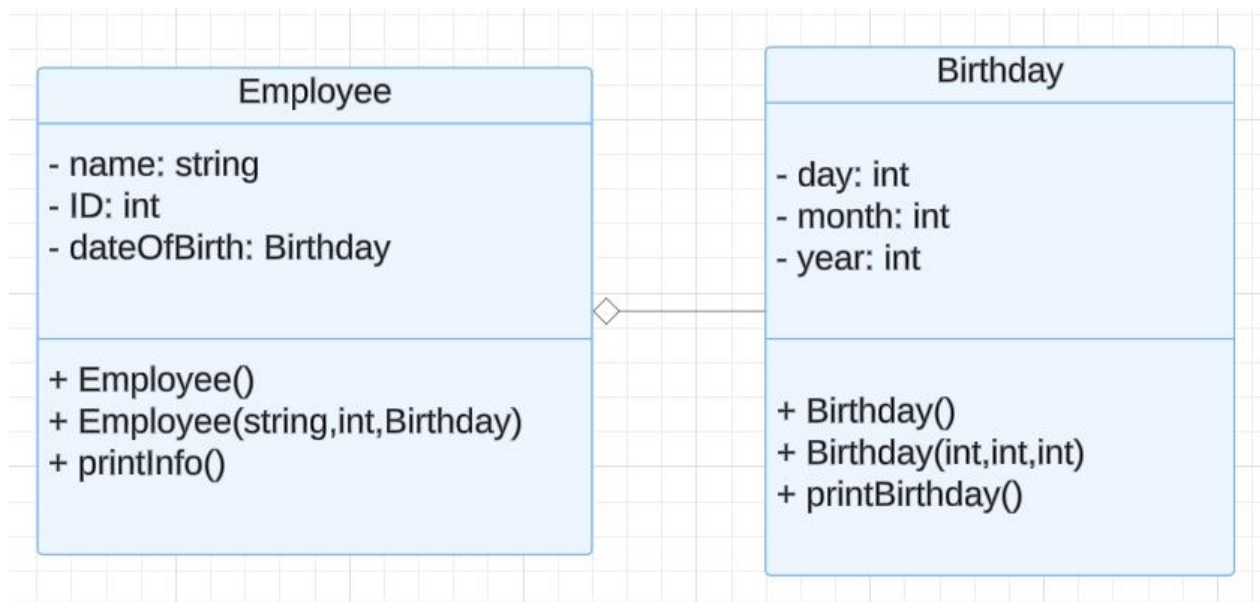
1. Describe what a virtual member function is.
2. A virtual function is re-defined in which of the following class?
 - a. Parent class
 - b. Derived class
 - c. Both A and B
3. Late binding can be implemented by which of the following?
 - a. Function overloading
 - b. Function overriding
 - c. Virtual functions
 - d. Both B and C
4. What will be the output of the following program?
 - a. 123
 - b. 12
 - c. 213
 - d. 321

```
1  #include <iostream>
2  using namespace std;
3  class BaseClass{
4  public:
5      virtual void myFunction() { cout << "1"; }
6  };
7
8  class DerivedClass1 : public BaseClass{
9  public:
10     void myFunction() { cout << "2"; }
11 };
12
13 class DerivedClass2 : public DerivedClass1{
14 public:
15     void myFunction() { cout << "3"; }
16 };
17 int main()
18 {
19     BaseClass* p;
20     BaseClass ob;
21     DerivedClass1 derivedObject1;
22     DerivedClass2 derivedObject2;
23     p = &ob;
24     p->myFunction();
25     p = &derivedObject1;
26     p->myFunction();
27     p = &derivedObject2;
28     p->myFunction();
29     return 0;
30 }
```

Object Oriented Programming Lab#8

5. Which of the following cannot be used for polymorphism?
- Overloaded member functions
 - Static member functions
 - Inline member functions

Problem 1 (Aggregation Example):



Demonstrate the classes in a program that:

PrintBirthday function: should not accept any arguments. It should have a cout statement that prints the birthday details in this order (`cout << day << " " << month << " " << year;`).

printInfo function: should not accept any arguments. It should have a cout statement that prints the name of the employee as along as his birthday (Hint: call the function named `printBirthday()`).

Main class:

- Create birthday object named **bd** using the parameterized constructor (day = 25, month = 5, year = 2002)
- Create an employee object named **emp1** using the parameterized constructor (name = "Hesham", ID = 1256, birthday object = bd).
- Call the `printInfo()` function for `emp1`.

Microsoft Visual Studio Debug Console

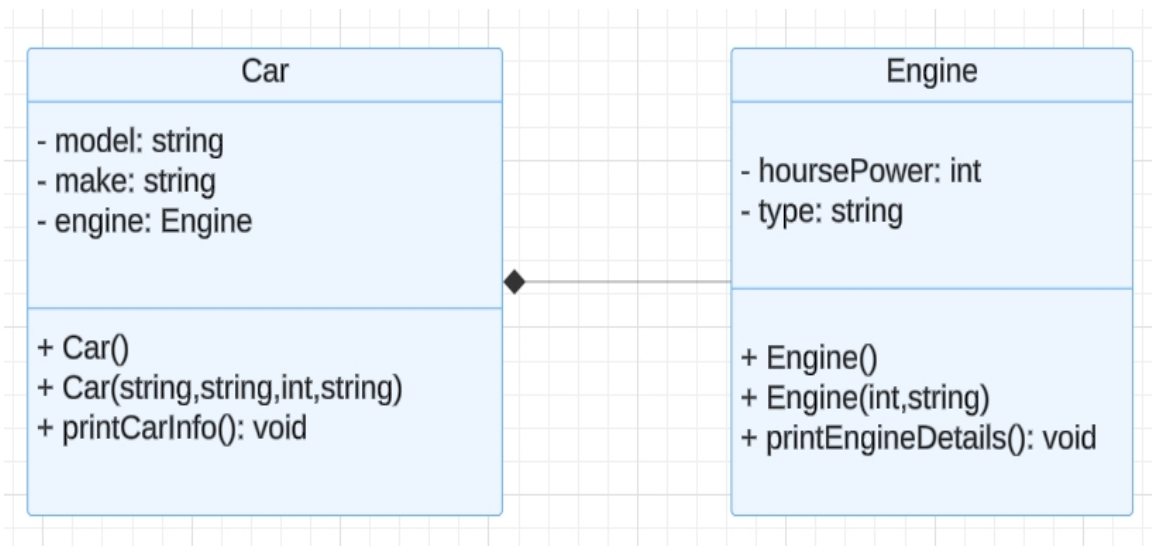
```
The employee name is: Hesham
and his birthday is on: 25/ 5/ 2002
```

```
C:\Users\DELL\source\repos\ConsoleApplication8\x64\Debug\AyahOOPLAB.exe (process 39460) exited with code 0.
```

```
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

Problem 2 Sample run

Problem 2 (Composition Example):



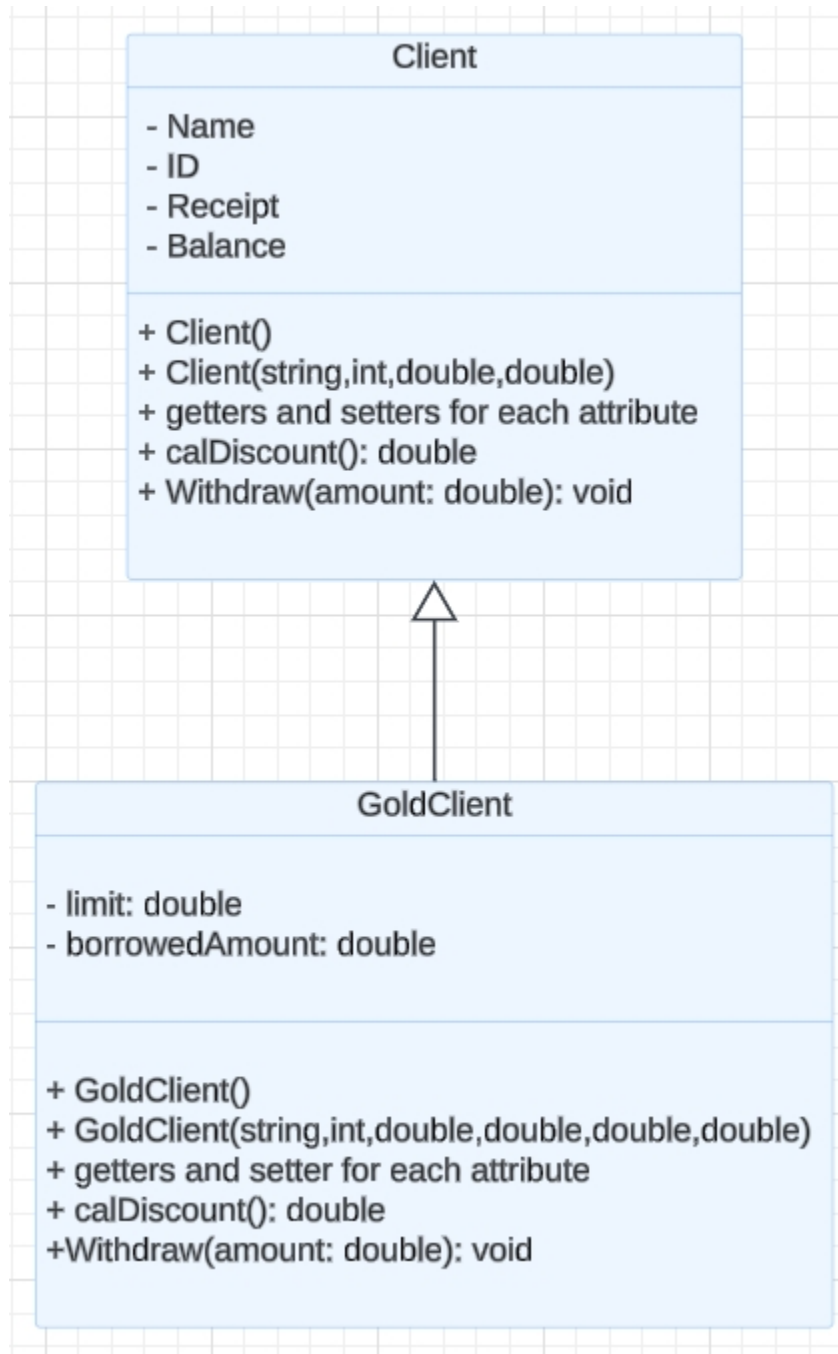
Demonstrate the classes in a program that:

1. **printEngineDetails** function: should not accept any arguments. It should print the engine's horsepower and type (e.g., `cout << horsepower << " HP " << type;`).
2. **printCarInfo** function: should not accept any arguments. It should print the car's model, make, and call `printEngineDetails()` to display engine details.

Main class:

1. Create a **Car** object named `car1` using the parameterized constructor (model = "Mustang", make = "Ford", horsepower = 450, type = "V8").
2. Call the `printCarInfo()` function for `car1`.

Problem 3 (Polymorphism Example):



Requirements:

For client class:

1. calDiscount(): should return the receipt after deducting 10% of its value.
(virtual)
2. Withdraw(): function that takes the desired withdraw amount as a parameter. It checks if the desired amount is less than or equal the client's balance or not. If the condition is satisfied the balance amount should be updated and "You have successfully withdrawn the desired amount- As a debit" should be displayed else, it will print "Insufficient Balance".
(virtual)


For GoldClient Class:

1. calDiscount(): Override the calDiscount() in the parent class that should return the receipt after deducting 20% of its value.
2. Withdraw (): function that takes the desired withdraw amount as a parameter. It checks whether the desired amount is less than or equal to the client's balance. If the condition is satisfied, then the function should update the balance amount and display the following "You have successfully withdrawn the desired amount- As a debit". If not and since the Gold Client has more privileges than the regular client, the client can borrow the desired amount if it doesn't exceed the limit and thus it will update the borrowed amount and limit value and displays the following "You have successfully withdrawn the desired amount based on your limit- As a credit".

Object Oriented Programming Lab#8

Demonstrate the classes by writing a program:

1. Create a Client object named C that has the name = "Sarah", ID = 9867, receipt = 980.5, balance = 11500.
2. Call the function calDiscount and print the result.
3. Call the withdraw function with an amount of 11,000 for the object named C and print the result.
4. Call the withdraw function again but with an amount of 580 for the object named C and print the result.
5. Create a GoldClient object named GC that has the name = "Moaz", ID = 8965, receipt = 980.75, balance = 11,500, limit = 5000.
6. Call the function calDiscount and print the result.
7. Call the withdraw function with an amount of 11,000 for the object named GC and print the result.
8. Call the withdraw function again but with an amount of 580 for the object named GC and print the result.
9. Create a client object named p (pointers) that has the address of GC.
10. Call the function calDiscount and print the result.
11. Call the withdraw function with an amount of 600 for the object named p and print the result. (Hint: use ->)

 Microsoft Visual Studio Debug Console

```
Discount for Client
882.45
You have successfully withdrawn the desired amount
Insufficient Balance !
Discount for GoldClient
784.4
You have successfully withdrawn the desired amount- As a debit
You have successfully withdrawn the desired amount based on your limit- As a credit

Discount for GoldClient
627.52
You have successfully withdrawn the desired amount based on your limit- As a credit
```