Object Oriented Programming – Fall 22
(BS-CS-F22)
Lab-11 (Polymorphism)
Lab Instructor: Mam Sanam Ahmad

Instructions:

- **Indent your code properly.**
- **Use meaningful variable and function names.**
- **Use the camelCase notation.**
- **Use meaningful prompt lines/labels for all input/output.**
- **Do NOT use any GLOBAL variable(s). However, global named constants may be used.**
- **This is an individual lab, you are strictly NOT allowed to discuss your solution with fellow colleagues, even not allowed to ask how is he/she is doing, it may result in negative marking. You can ONLY discuss with your TAs or with me. • Anyone caught in an act of plagiarism would be awarded an "F" grade in this Lab.**

Do Validations on inputs where required otherwise 1 mark will be deducted for every wrong validation.

Total Marks 40

Task 01: [20 Marks]

Design a Ship class that has the following members:

- A member variable for the name of the ship (a string)
- A member variable for the year that the ship was built (a string)
- A constructor and appropriate accessors and mutators
- A virtual print function that displays the ship's name and the year it was built.

Design a **CruiseShip** class that is derived from the Ship class. The CruiseShip class should have the following members:

- A member variable for the maximum number of passengers (an int)
  A constructor and appropriate accessors and mutators
- A print function that overrides the print function in the base class. The

CruiseShip class's print function should display only the ship's name and the maximum number of passengers.

Design a **CargoShip** class that is derived from the Ship class. The CargoShip class should have the following members:

- A member variable for the cargo capacity in tonnage (an int). ▪ A constructor and appropriate accessors and mutators.
- A print function that overrides the print function in the base class. The CargoShip class's print function should display only the ship's name and the ship's cargo capacity.

Design another class named **BattleShip** that is derived from the Ship class. The BattleShip class should have the following members:

- A member variable for the total number of missiles (an int).
- Appropriate constructors, acessors and mutators.

- A print function that overrides the print function in the base class. The BattleShip class's print function should display only the ship's name and the missiles' capacity.

Demonstrate the classes in a program that has an array of Ship pointers. The array elements should be initialized with the addresses of dynamically allocated Ship CruiseShip, and CargoShip objects (Ask user which ships he/she wants to create). The program should then step through the array, calling each object's print function.

Task 02: [20 Marks]

Create a program that simulates a bank account system. The program should have the following features:

- The bank has different types of accounts: savings account and checking account.
- Each account has a unique account number, balance, and interest rate.
- The program should allow the user to create a new account, deposit or withdraw money from an existing account, calculate the interest earned on an account, and display the current balance and account information.
- The interest rate for each account type should be set by the bank and should be a virtual function in the account class, so that it can be overridden by each account type.
- The program should also have a function to display the list of accounts owned by a particular customer.

The program should have the following classes:

- **Account:** Represents a bank account with a unique account number, balance, and interest rate. Has functions to deposit or withdraw money, calculate the interest earned, and display the account information. Also has a virtual function to calculate the interest rate.
- **SavingsAccount:** Represents a savings account, which has a higher interest rate than a checking account. Overrides the virtual function to set the interest rate. This interest rate should be calculated according to the new rate and this should be more than 20%.
- **CheckingAccount:** Represents a checking account, which has a lower interest rate than a savings account. Overrides the virtual function to set the interest rate. This interest rate should be less than 10%.
- **Customer:** Represents a customer with a unique ID and a list of accounts they own. Has functions to create accounts, deposit or withdraw money from accounts, and display the customer information. Also has a function to display the list of accounts owned by the customer.