MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS
CNG443 – Worksheet for Lecture 4

Purpose:

In this work sheet, you will get hands-on experience with inheritance and polymorphism.

Tasks:

1. In Worksheet 3 (previous worksheet), you created an Account class. That Account class was defined to model a bank account. An account has the properties account number, balance, annual interest rate, and date created, and methods to deposit and withdraw funds. Create two subclasses for checking and saving accounts. A checking account has an overdraft limit, but a savings account cannot be overdrawn.
   Draw the UML diagram for the classes and then implement them. Write a test program that creates objects of Account, SavingsAccount, and CheckingAccount and invokes their toString() methods.
2. Use the class hierarchy created in the first exercise to simulate an ATM machine. In this ATM machine, you will create 10 accounts where 3 of them are Savings account, 3 of them are Checking accounts and the rest is accounts without any further details. All these accounts have an initial balance 50 dollars. Keep these accunts in a single Arraylist and write an ATM class that prompts the user to enter an id. If the id is entered incorrectly, ask the user to enter a correct id. Once an id is accepted, display a menu as follows.
   Sample run:

```
Enter an account id: 3

Main menu:
1: check balance
2: withdraw
3: deposit
4: exit
Enter a choice: 1
The balance is 50!

Main menu:
1: check balance
2: withdraw
3: deposit
4: exit
Enter a choice: 2
Enter an amount to withdraw: 25
...
```

3. In this exercise you will create an arraylist that holds the following objects together: Loan object, a Date object, a String object, a Random object, and a GraphObject object. You will write a program that keeps these objects in the same array list and displays their string representation. To display their String representation, you need to use object's toString method. Class definitions are as follows:
   - Loan class has private fields of annualInterestRate, numberOfYears, loanAmount and loanDate with the type of java.util.Date. You need to implement the necessary getters and setters, and at least two constructors.
   - Date – use the java.util.Date
   - String – use java.lang.String
   - Random – use java.util.Random
   - GraphObject class represents a shape that has the following private fields: color, x-coordinate, y-coordinate, caption. You need to implement necessary getters and setters and at least two constructors.