# Introduction to Computer Science 2

## Lab 3: Inheritance

### Learning Goals:

- To learn the principle of inheritance.
- To learn how the principle of inheritance allows us to reuse code.
- To learn how to extend existing and new classes.

### Exercise 1 (3 points)

Add a `TimeDepositAccount` class to the bank account hierarchy. The time deposit account is just like a savings account, but you promise to leave the money in the account for a particular number of months. There is a penalty for early withdrawal. Construct the account with the interest rate and the number of months to maturity. In the `addInterest` method, decrement the count of months. If the count is positive during a withdrawal, charge the withdrawal penalty.

### Exercise 2 (3 points)

Implement a subclass `Square` that extends the `Rectangle` class. In the constructor, accept the x- and y-positions of the center and the side length of the square. Call the `setLocation` and `setSize` methods of the `Rectangle` class. Look up these methods in the documentation for the `Rectangle` class. Also supply a method `getArea` that computes and returns the area of the square. Write a sample program that asks for the center and side length, then prints out the square (using the `toString` method that you inherit from `Rectangle`) and the area of the square.

### Exercise 3 (4 points)

Make a class `Employee` with a name and salary. Make a class `Manager` inherit from `Employee`. Add an instance field, named department, of type `String`. Supply a method `toString` that prints the manager's name, department, and salary. Make a class `Executive` inherit from `Manager`. Supply appropriate `toString` methods for all classes. Supply a test program that tests these classes and methods.

## Honor code, coding style, and deliverable:

Try to solve the exercises with what you already know. You are welcome to expand your program to do extra things but they are not mandatory.

**Plagiarism is not allowed!** We will run sophisticated software that automatically detects similarities on source code among students. All plagiarism incidents will be immediately reported to the Board of Examiners!

## Submission!

**Submit your java files to canvas.**
**Ask your instructor in case there is a problem with your submission.**

**DO NOT SEND SUBMISSIONS VIA EMAIL**
**YOUR LAB WILL NOT GET GRADED!**