

Lab 2

Objectives

- To practice overriding data structures.
- To learn more about Classes.

Code quality

The code you submit should be easily readable, clean and well-commented.

Commented-out sections of non-working code should be removed. Indentations and white spaces should be used to separate

logical blocks of the code. Variables, methods and classes should have descriptive names.

Every class and method should have a comment according to Javadoc format (see Zybook).

When overriding methods, use `@Override` compiler annotation.

Int and Long

Some information:

- An Int is a signed 32-bit type that has a range from $-2,147,483,648$ to $2,147,483,647$.
 - A long is a signed 64-bit type and is useful for those occasions where an int type is not large enough to hold the desired value. It has a minimum value of $-9,223,372,036,854,775,808$ and a maximum value of $9,223,372,036,854,775,807$ (inclusive).
1. Create a class myLong that is capable of manipulating integers up to size 100 digits! As a guideline, you can get help from arrays to store your values and implement this idea. Or in any other ways that you find it more convenient to work with.
For example: the number 3,125,311,345,678,998 should get store in array
 $a = [312531, 1345678998]$
 2. Create method setLong to get a long integer from key board.
 3. Create a method getLong that displays your long values on monitor.
 4. Create methods add, subtract and multiply to add, subtract and multiply longs and return the value as an array.

Keep in mind that you are not allowed to use any build In methods that convert integer into longs or vice versa.