

GTU Department of Computer Engineering
CSE 222/505 - Spring 2022

Homework 1

Due date: March 7 2022– 09:00 AM

This homework is designed for you

- to review OOP concepts and Java programming,
- to get used to reporting conventions that will be used in the course.

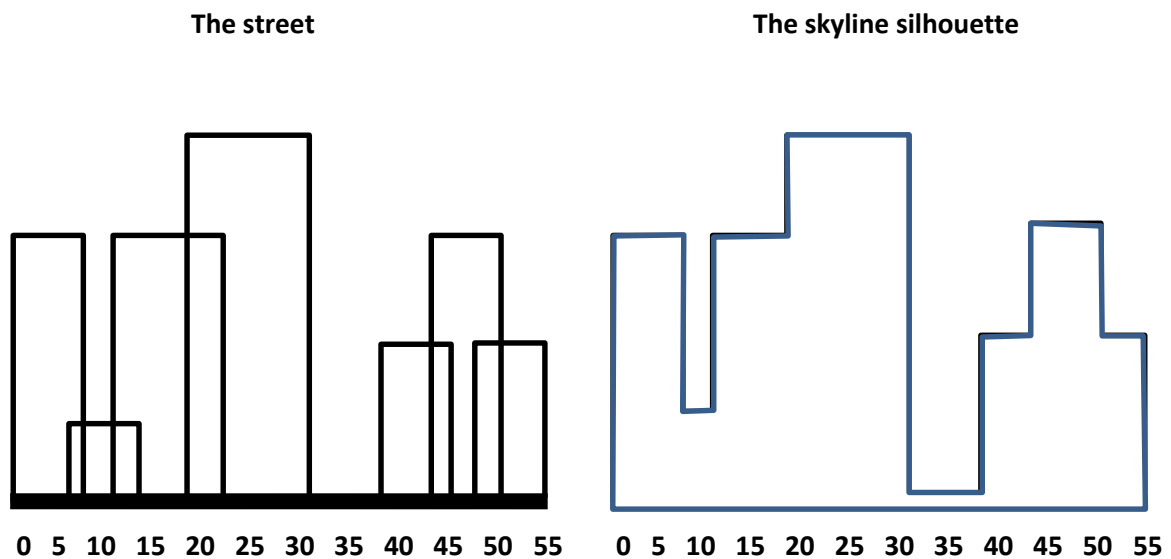
In this homework, you are not allowed to use any data structure other than simple arrays. Later, we may reuse the same scenario and ask you to implement the same system using some data structures introduced during this course.

SCENARIO:

You are asked to design and implement a city planning software that will be used for designing a small one street town. The software should meet the following requirements:

- The length of street can be set by the user.
- The street has two sides and both sides are designed.
- The street may contain three types of buildings (houses, offices, markets) and playgrounds. Please note, it is not necessary that all kinds of buildings are available on the street.
- The position, length (the length on the street, the depth of the buildings is not considered) and height of the buildings are set by the user. A building can be placed at a land on the street only if there is enough land for that building in terms of length.
- Besides the length and height properties of the building, there are some type-specific properties:
 - The houses have the number of rooms, color, and owner,
 - The offices have job-type and owner,
 - The markets have owner and opening/closing times properties.
- The software enables focusing on a building. Different types of information are presented to the user on focusing on a building:
 - The house presents its owner,
 - The office presents its job types,
 - The playground presents its length,
 - The market presents its closing time.
- There are two different modes for using the program: Editing mode and viewing mode.
- In editing mode, the user can:
 - add/delete a building on a land in the street.

- In viewing mode, the user can:
 - display the total remaining length of lands on the street.
 - display the list of buildings on the street.
 - display the number and ratio of length of playgrounds in the street.
 - calculate the total length of street occupied by the markets, houses or offices.
 - display the skyline silhouette of the street (Please see the figure below).



RESTRICTIONS:

- Use only Arrays, any other data type is prohibited.
- Use only one main class to drive the others from that superclass.
- Don't use any other third-party library.

GENERAL RULES:

- For any question firstly use [the forum](#) on the MS Teams page, and then the contact TA.
- You can submit an assignment one day late and will be evaluated over sixty percent (%60).

TECHNICAL RULES:

- You must write a driver function that demonstrates all possible actions in your homework. For example, if you are asked to implement an array list and perform an iterative search on the list then, you must at least provide the following in the driver function:
 - Create an array list and add items to the list. Append items to the head, tail, and k^{th} index of the list.
 - Perform at least two different searches by using two items in the list and print the index of the items.
 - Perform another search with an item that isn't in the array list and inform the user that the item doesn't exist in the array list.

- Delete an existing item from the list and repeat the searches.
- Try to delete an item that is not on the array list and throw an exception for this situation.

The driver function should run when the code file is executed.

- Implement [clean code standards](#) in your code;
 - Classes, methods, and variables names must be meaningful and related to the functionality.
 - Your functions and classes must be simple, general, reusable, and focus on one topic.
 - Use standard [java code name conventions](#).

REPORT RULES:

- Add all [javadoc](#) documentations for classes, methods, variables ...etc. All explanations must be meaningful and understandable.
- You should submit your homework code, Javadoc, and report to MS Teams in a "studentid_hw1.tar.gz" file.
- Use the given homework format including [selected parts from the table below](#):

Detailed system requirements	✓
The Project use case diagrams (extra points)	✓
Class diagrams	✓
Other diagrams	X
Problem solutions approach	✓
Test cases	✓
Running command and results	✓

GRADING :

- **No OOP design:** -100
- **No method overriding:** -95
- **No error handling:** -50
- **No inheritance:** -95
- **No polymorphism:** -95
- No javadoc documentation: -50
- No report: -90
- Disobey restrictions: -100
- **Cheating:** -200
- Your solution is evaluated over 100 as your performance.

CONTACT :

- Teaching Assistant M. Burak Koca, b.koca@gtu.edu.tr