

# CSE 112 2. Midterm

Olcay Taner YILDIZ

## I. QUESTION (3 POINTS)

Design enumerated class named FlatType to represent flat types. The class Flat contains NORMAL, DUBLEX, BASEMENT.

## II. QUESTION (14 POINTS)

Design class named Flat to represent flats. The class Flat contains:

- (1 pt) A string data field named **owner** that specify the owner of the flat.
- (1 pt) An integer data field named **roomCount** that specify the number of rooms the flat has.
- (1 pt) An integer data field named **area** that specify the area of the flat.
- (1 pt) An FlatType data field named **flatType** that specify the type of the flat.
- (4 pts) Four argument constructor **Flat(String newOwner, FlatType flatType, int roomCount, int area)**.
- (6 pts) Four getter methods for the four fields **owner, roomCount, area, flatType**.

## III. QUESTION (26 POINTS)

Design class named Building to represent buildings in a city. The class Building contains:

- (1 pt) A string data field named **name** that specify the name of the building.
- (1 pt) An integer data field named **flatCount** that specify the number of flats the building has.
- (1 pt) A boolean data field named **withLift** that specify if the building has a lift.
- (1 pt) An array (not an arraylist) of flat named **flats** that specify the flats the building has.
- (3 pts) Two argument constructor **Building(String name, boolean withLift)**.
- (6 pts) A method to remove a flat with a given owner name from the building **void remove(String ownerName)**.
- (2 pts) A method to add a flat to the building **void add(Flat flat)**.
- (4 pts) A method to find the total area of the building **int totalArea()**
- (5 pts) Three getter methods for the three fields **name, flats, withLift**.
- (2 pts) Method **Flat getFlat(int idx)** which returns the flat with index idx.

## IV. QUESTION (12 POINTS)

Design class named Street to represent streets in a city. The class Street contains:

- (1 pt) A string data field named **name** that specify the name of the street.
- (1 pt) A method to add a building to the street **void add(Building building)**.
- (4 pts) A method to remove a building with a given name from the street **void remove(String buildingName)**.
- (1 pt) An arraylist (not an ordinary array) of building named **buildings** that specify the buildings of the street.

- (2 pts) One argument constructor **Street(String name)**.
- (1 pt) One getter method for the field **name**.
- (2 pts) Method **Building getBuilding(int idx)** which returns the building with index idx.

## V. QUESTION (12 POINTS)

Write a main class Test using previous Questions to do the following:

- (3 pts) Declare three Flats one owned by Seda, one owned by Olcay, and the other owned by Sezin where Seda lives in a flat with 4 rooms of  $140m^2$  of dublex type, Sezin lives in a flat of basement type with 3 rooms of  $110m^2$  and Olcay lives in a normal flat with 2 rooms of  $100m^2$ .
- (3 pts) Declare a building named Cse112 with a lift where Seda and Olcay live.
- (2 pts) Declare a building named Cse202 with no lift where Sezin lives.
- (3 pts) Declare a street named Cse where there are only two buildings namely Cse112 and Cse202.
- (1 pt) Display the total area of building Cse112.

## VI. QUESTION (11 POINTS)

Write a function double totalValue() in the class Street which calculates the total value (in TL) of the street with the following constraints:

- The value of a basement flat is calculated as  $area * 1500$ , normal flat is calculated as  $area * 2000$  and the value of the flat of roof type is calculated as  $area * 2500$ .
- The value of a flat increases by 20 percent if it is in a building with a lift.

## VII. QUESTION (11 POINTS)

Write a function void printBuilding(String fileName) in the class Building which prints all information of a building to the file with name fileName. Example output file for Cse112 is:

```
Cse112
2 true
Seda 4 140 dublex
Olcay 2 100 normal
```

For Cse202, it is

```
Cse202
1 false
Sezin 3 110 basement
```

## VIII. QUESTION (11 POINTS)

Write a constructor function Building(String fileName) in the class Building which creates a new building from the file with name fileName.