Assignment 5

**Problem statement**: This program will stare 9 items in a 2-dimensional array with each item being a structure value. The user must choose a treasure chest size that is more than 1 to allow the storage of at least 1 item. The items stored in the treasure chests are hotel keys, the floor it is on, the number of beds in the room, the cost of the room per night, the existence of TV in the room. The 6 functions are as follow:

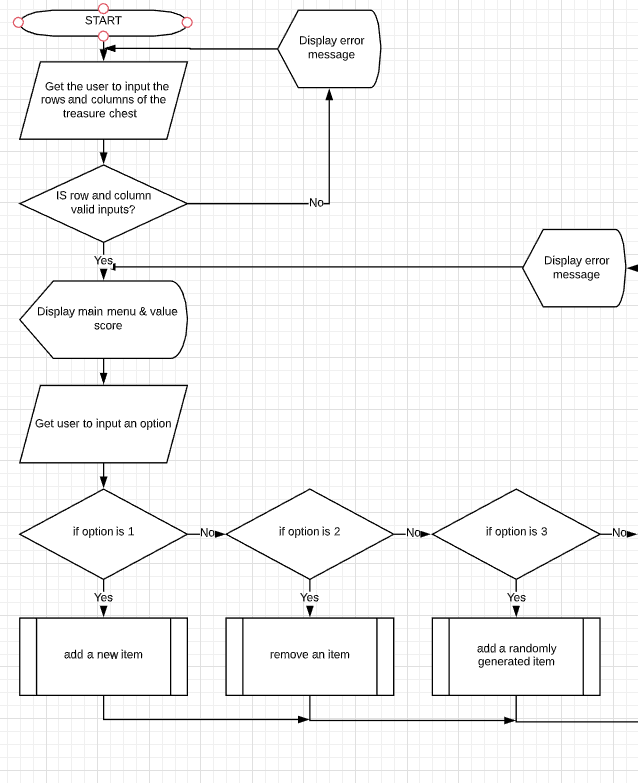
1. Add an item of the user’s choice where the attribute is specified by me.
   1. Floor number – The floor number has to be a whole number and it can be in the basement(taken as a negative number).
   2. Number of beds – The room cannot have a negative number of beds.
   3. Cost of room per night– The room has to have a value of at least 1 per night.
   4. Existence of TV – Boolean value of true or false.

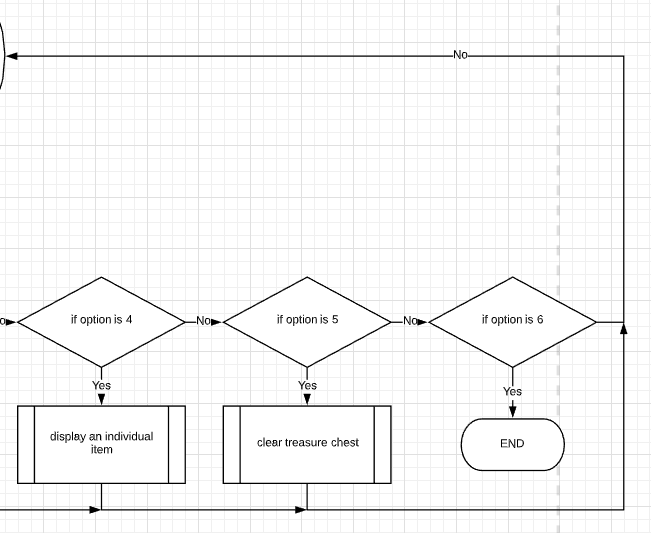
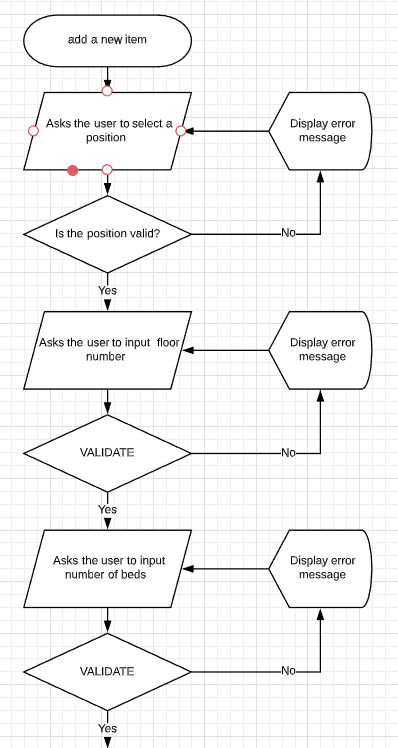
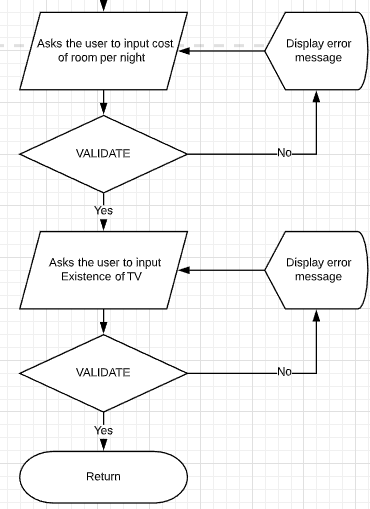
If the item exists at the location, prompt the user it is already there; if the user wants to add an item where it is out of range, prompt the user it is out of range.

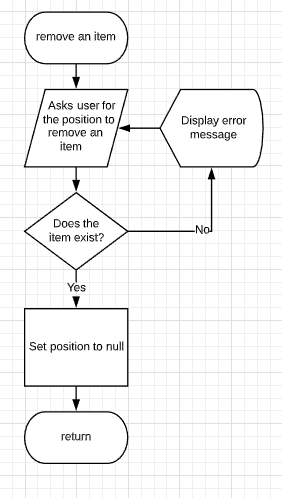
1. Remove an item from the chest if and only if the item exists at the spot. Else prompt an error message.
2. Randomly generate values for every attribute in the treasure chest spot:
   1. Floor number – Random int positive / negative
   2. Number of beds – random positive integer
   3. Costs of room – random positive float(not 0)
   4. Existence of TV – Random true or false value
3. Shows the item at the location given that the item exists. Item doesn’t exist then prompt the user; item locating is not within the boundary set then prompt the user.
4. Clear every item in the treasure chest!
5. Exit the program.

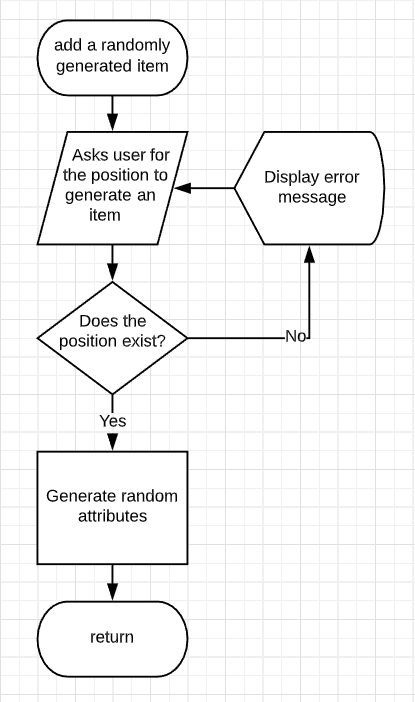
I assume that the user will enter int value for options in the menu, and so forth as indicated in the context.

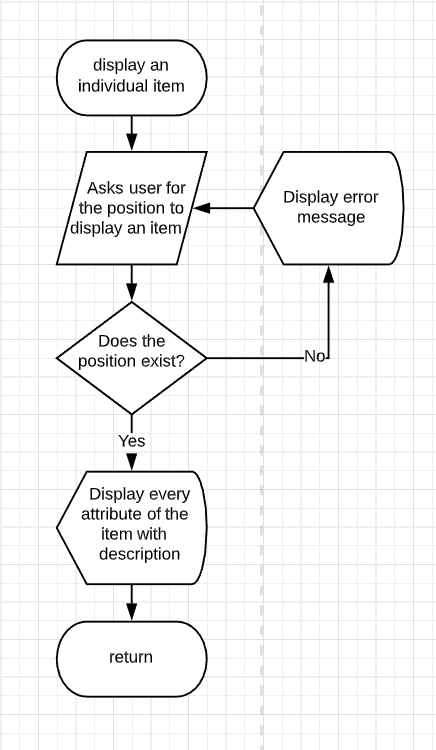
**Devise a plan**

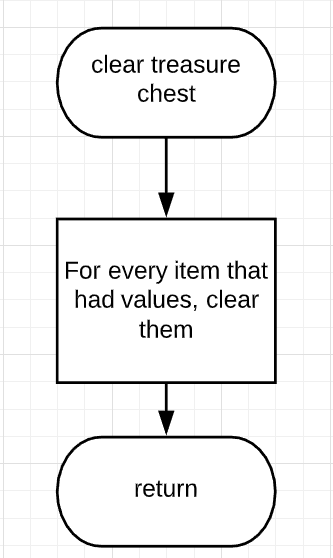
****









**Test cases**

|  |  |  |
| --- | --- | --- |
| **Test Case setting** | **User input** | **Expected result** |
| In a 3x3 treasure chest, [1][1] contains an item  User chooses to remove an item | Row:1  Column:4 | Output “The location [1][4] does not exist within the treasure chest!” |
| In a 3x3 treasure chest, [1][1] contains an item  User chooses to add an item | Row:1  Column:1 | Outputs “The position already has an item and cannot add another item in it.” |
| In a 3x3 treasure chest  User chooses to add an item | Row:3  Column:1 | Continues to the index [2][0] to allow the user to add an item |
| In a 3x3 treasure chest, [1][1] contains an item  User chooses to generate an item | Row:1  Column:1 | Outputs “The position already has an item and cannot add another item in it.” |
| In a 3x3 treasure chest  User chooses to generate an item | Row:2  Column:1 | Generate an item at location [1][0] |
| In a 3x3 treasure chest  User chooses to generate an item | Row:2  Column:4 | Output “The location [2][4] does not exist within the treasure chest!” |
| In a 3x3 treasure chest, only the [1][1] location contains an item  User chooses to show an item | Row:1  Column:2 | Output “The location doesn’t have an item in it.” |
| In a 3x3 treasure chest  User chooses to show an item | Row:2  Column:4 | Output “The location [2][4] does not exist within the treasure chest!” |
| How many rows do you want in your chest | 0 | The treasure chest has to have at least 1 row |