SC1003 2022/23 Sem 1 Assignment

The NTU Hotel has a capacity of MAX (e.g., 5 for this program) rooms. You are required to write a NTU Hotel Room Reservation Program. The program uses an array of MAX structures for hotel rooms.

Each structure should hold:

- roomID – It stores the room identification number.
- status – It stores a marker that indicates whether the room is assigned or not (i.e., EMPTY or TAKEN).
- customerName – It stores the name of the customer who occupies the room.

A structure is defined to represent a Room record of a Hotel as follows:

```
typedef struct
{
   int roomID;
   int status;
   char customerName[20];
} Room;
```

The **roomID** must be **unique**. Therefore, each Room structure will have a different number of roomID when it is created. You may assume that the name of a customer is not more than 20 characters long. In the program, it should display a menu to support the following functions:

NTU HOTEL ROOM RESERVATION PROGRAM:
1: listOccupiedRooms()
2: assignRoom()
3: removeRoom()
4: findCustomer()
5: quit

The program should execute the functions (or options) of its menu. It will continue execution until the user selects to quit from the program.

The functions are described as follows:

(1) listOccupiedRooms() – The function prints the list of room assignments in hotel.

Please note:
- The function prints the following message when it is executed: **"listOccupiedRooms():"**
- If the hotel is empty, i.e., no room is occupied, the function should display the message: **"The hotel is empty"**.

(2) assignRoom() – The function assigns a customer to a room. It reads in the selected room number (roomID) and customer name from user, and then assigns the room accordingly.

Please note:
- The function prints the following message when it is executed: **"assignRoom():"**
- After assigning a room suceessfully, the function should display the message: **"The room has been assigned successfully "**.

- If the hotel is full (i.e. MAX) during room assignment, the function should display the message: **"The hotel is full"**.
- If the selected room has been assigned to another customer already, the function should display the message: **"Occupied! Enter another roomID"**.
- If the selected room number is not between 1 and MAX (i.e., 5), the function should display the message: **"Enter a roomID between 1 and 5"**. The function should continue to read a room number from the user.

(3) removeRoom() – The function removes a room assignment. It reads in the selected room number (i.e. roomID) from user and then remove the assigned room accordingly.

Please note:
- The function prints the following message when it is executed: **"removeRoom():"**
- After room removal, the function should display the message: **"Removal is successful"**.
- If all the rooms in the hotel are empty, the function should display the message: **"All the rooms are empty"**.
- If the selected room is empty, the function should display the message: **"Empty! Enter another roomID for removal"**.
- If the selected room number is not between 1 and 5 (i.e., MAX), the function should display the message: **"Enter a roomID between 1 and 5"**.

(4) findCustomer() – The function finds the target customer name from the Room records stored in hotel, and prints the target room information to the screen.

Please note:
- The function prints the following message when it is executed: **"findCustomer():"**
- The function finds the **first appearance** of the target customer name in the array structure (i.e. according to roomID), and prints the corresponding target customer information to the screen.
- The target string matching should be done irregardless of the letter cases (upper case letter and lower case letter will be treated to be the same).
- If target customer name is found in hotel, the function should display the message: **"The target customer name is found"** and display the room information of the corresponding customer.
- If target customer name does not exist in hotel, the function should display the message: **"The target customer name is not found"**.

You are required to write the program and the functions according to the program requirements. Note that:

- You should design the program and define the functions according to the requirements.
- You only need to consider the requirements stated in the program specification, and you do not need to implement any user input checking which is not stated in the specification.
- You may add any other supporting functions in the program if needed.
- You may include any C library functions in your program if needed.
- Sample test case are given below.

A sample program running session is given below (please note that the input data are shown in orange color):

NTU HOTEL ROOM RESERVATION PROGRAM:
1: listOccupiedRooms()
2: assignRoom()
3: removeRoom()
4: findCustomer()
5: quit
Enter your choice:
1
listOccupiedRooms():
The hotel is empty
Enter your choice:
2
assignRoom():
Enter a roomID between 1 and 5:
1
Enter customer name:
SC Hui
The room has been assigned successfully
Enter your choice:
1
listOccupiedRooms():
roomID: 1
customer name: SC Hui
Enter your choice:
5

The test cases for the program are given below. The test cases are organized in the form of input and output data for the convenience of your testing in APAS.

Sample Test Cases (Pretest)

Case 1 – assignRoom (3 rooms) and listOccupiedRooms

Input	Output
1 2 2 PC Leong 2 4 Daniel Foo 2 3 WC Tan 1 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: listOccupiedRooms(): **The hotel is empty** Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom():

	Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: listOccupiedRooms(): roomID: 2 customer name: PC Leong roomID: 3 customer name: WC Tan roomID: 4 customer name: Daniel Foo Enter your choice:

Case 2 – assignRoom (3 rooms and existed)

Input	**Output**
2 2 PC Leong 2 3 WC Tan 2 8 4 Daniel Foo 1 2 2 5 SC Hui 1 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: listOccupiedRooms(): roomID: 2 customer name: PC Leong roomID: 3 customer name: WC Tan roomID: 4 customer name: Daniel Foo Enter your choice: assignRoom(): Enter a roomID between 1 and 5: **Occupied! Enter another roomID** Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: listOccupiedRooms(): roomID: 2 customer name: PC Leong roomID: 3 customer name: WC Tan roomID: 4

	customer name: Daniel Foo roomID: 5 customer name: SC Hui Enter your choice:

Case 3 – assignRoom (full condition)

Input	Output
2 1 SC Hui 2 3 PC Leong 2 5 WC Tan 2 2 Philip Fu 2 4 Daniel Foo 1 2 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: listOccupiedRooms(): roomID: 1 customer name: SC Hui roomID: 2 customer name: Philip Fu roomID: 3 customer name: PC Leong roomID: 4 customer name: Daniel Foo roomID: 5 customer name: WC Tan Enter your choice: assignRoom(): **The hotel is full** Enter your choice:

Case 4 – removeRoom (empty and then roomID found)

Input	Output

1 3 2 2 PC Leong 2 3 WC Tan 2 4 Daniel Foo 1 3 3 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: listOccupiedRooms(): The hotel is empty Enter your choice: removeRoom(): **All the rooms are empty** Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: listOccupiedRooms(): roomID: 2 customer name: PC Leong roomID: 3 customer name: WC Tan roomID: 4 customer name: Daniel Foo Enter your choice: removeRoom(): Enter a roomID between 1 and 5: Removal is successful Enter your choice:

Case 5 – removeRoom (roomID found)

Input	**Output**
2 2 PC Leong 2 3 WC Tan 2 4 Daniel Foo 1 3 7 3 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully

	Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: listOccupiedRooms(): roomID: 2 customer name: PC Leong roomID: 3 customer name: WC Tan roomID: 4 customer name: Daniel Foo Enter your choice: removeRoom(): Enter a roomID between 1 and 5: Enter a roomID between 1 and 5: Removal is successful Enter your choice:

Case 6 – removeRoom (roomID not found)

Input	**Output**
2 2 PC Leong 2 3 WC Tan 2 4 Daniel Foo 1 3 1 4 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: listOccupiedRooms(): roomID: 2 customer name: PC Leong roomID: 3 customer name: WC Tan roomID: 4 customer name: Daniel Foo Enter your choice: removeRoom(): Enter a roomID between 1 and 5: **Empty! Enter another roomID for removal** Enter a roomID between 1 and 5: Removal is successful Enter your choice:

Case 7 – removeRoom (all empty rooms)

Input	Output
2 2 PC Leong 2 3 WC Tan 2 4 Daniel Foo 3 3 3 4 3 2 3 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: removeRoom(): Enter a roomID between 1 and 5: Removal is successful Enter your choice: removeRoom(): Enter a roomID between 1 and 5: Removal is successful Enter your choice: removeRoom(): Enter a roomID between 1 and 5: Removal is successful Enter your choice: removeRoom(): **All the rooms are empty** Enter your choice:

Case 8 – findCustomer (taget name found)

Input	Output
2 2 PC Leong 2 3 WC Tan 2 4 Daniel Foo 4 WC Tan 4 wc Tan 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5:

	Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: findCustomer(): Enter customer name: The target customer name is found roomID: 3 customer name: WC Tan Enter your choice: findCustomer(): Enter customer name: **The target customer name is found** roomID: 3 customer name: WC Tan Enter your choice:

Case 9 – findCustomer (target name not found)

Input	**Output**
2 2 PC Leong 2 3 WC Tan 2 4 Daniel Foo 4 pcc Leong 5	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer() 5: quit Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: findCustomer(): Enter customer name: **The target customer name is not found** Enter your choice:

Case 10 – findCustomer (unique roomID with duplicated target names)

Input	**Output**
2 4 PC Leong 2 3	NTU HOTEL ROOM RESERVATION PROGRAM: 1: listOccupiedRooms() 2: assignRoom() 3: removeRoom() 4: findCustomer()

WC Tan 2 2 PC Leong 4 PC Leong 4 Pc leong 5	5: quit Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: assignRoom(): Enter a roomID between 1 and 5: Enter customer name: The room has been assigned successfully Enter your choice: findCustomer(): Enter customer name: **The target customer name is found** roomID: 2 customer name: PC Leong Enter your choice: findCustomer(): Enter customer name: **The target customer name is found** roomID: 2 customer name: PC Leong Enter your choice: