

Hospital class

The Hospital class is used to represent a hospital that can admit patients and manage their information.

Creating a Hospital instance

You can create a new instance of the Hospital class using the default constructor or the constructor that takes a number of available beds as an argument:

```
// Create a new hospital with no available beds
```

```
Hospital myHospital;
```

```
// Create a new hospital with 50 available beds
```

```
Hospital myHospital(50);
```

Adding a patient

To add a new patient to the hospital, use the `addPatient` method. This method takes a patient object as an argument and adds it to the head of a linked list representing the patients in the hospital.

```
patient newPatient("John Doe", "1234", 35, "stable");
```

```
myHospital.addPatient(newPatient);
```

If there are no available beds in the hospital, the method will print a message and return without adding the patient.

Checking the number of available beds

To check the number of available beds in the hospital, use the `checkNumBeds` method. This method will print a message indicating the number of available beds

```
myHospital.checkNumBeds(); // Prints "the available  
beds in the hospital are: 50"
```

Deleting a patient

To delete a patient from the hospital, use the `deletePatient` method. This method takes a patient ID as an argument and searches for the patient in the linked list. If the patient is found, it is removed from the list and the number of available beds is incremented.

```
myHospital.deletePatient("1234");
```

If the patient is not found in the list, the method will print a message indicating that the patient was not found.

Searching for patients

To search for patients in the hospital, use the `searchPatients` method. This method takes a search term as an argument and returns a vector of patient objects whose name or ID matches the search term.

```
vector<patient> results =
```

```
myHospital.searchPatients("John Doe");
```

Displaying all patients

To display information about all patients in the hospital, use the `displayPatients` method. This method will print the name, ID, age, and health status of each patient in the linked list.

```
myHospital.displayPatients();
```

Class members

`numBeds`

An integer representing the number of available beds in the hospital.

`head`

A pointer to the head node of a linked list representing the patients in the hospital.

Constructor methods

`Hospital()`

The default constructor for the `Hospital` class sets the number of available beds to 0 and initializes the head pointer to null.

`Hospital(int numBeds)`

A constructor for the `Hospital` class that takes a number of available beds as an argument and initializes the `numBeds` member variable to that value. The head pointer is initialized to null.

`getNumBeds()`

A const method that returns the current number of available beds in the hospital.

`addPatient(const patient& data)`

A method that adds a new patient to the hospital. If there are no available beds in the hospital, the method prints a message and returns without adding