CS110: Computer Programming Lab
Department of CSE
IIT, Guwahati

Module 04 Stage 02 Exercise 05

Assessment exercises are designed to help us check if the student has learned the basics of the topics included in the drill instructions. However, a drill assessment is not a comprehensive assessment. A fuller and complete assessment aimed at determining the course grades will be done through CS110 examinations.

It is expected that the student will attempt and solve many more exercises from the drill assessment sets to improve their programming skills and for an excellent performance at the examinations.

Exercise

Use the following `struct` as the front reference to a queue:

```
struct queue
{
    /* and the one who joined the earliest */
    struct queue_elem * ancientElemP;
    /* Size of queue */
    int size;
};
```

Note that the `struct` has reference only to the element in the linked-list that will leave the queue next. The list needs to be navigated to find its other end when needed.

Implement the queue based on the `struct` defined above. In your implementation change the specification of method `joinQ()`, It will only add a new entry if the reference to the application object being added to the queue is not already in the queue.

If the reference is already in the list, the queue will remain unchanged and the new size value of 0 is returned to signify that the action failed to add the entry.

Add new tests to tests-first file to test the correctness of the altered specifications (contract) of method `joinQ()`

Suggested Interface

```
#ifndef QUEUE_H_INCLUDED
#define QUEUE_H_INCLUDED

/* Returns a reference (pointer) to a new queue */
void * mkQueue(void);

/* Removes a queue specified by a valid reference */
int rmQueue (void *);

/* Returns number of entries in a validly referenced queue */
int sizeQ (void *);

/* Place objP  in queueP and returns new count of entries */
int joinQ (void * queueP, void * objP);

/* Return earliest arrived objP in queue */
void * leaveQ (void *);

#endif // QUEUE_H_INCLUDED
```