```
#include "queue1.h"
#include <stdlib.h>
#include <assert.h>

typedef struct list *list;

struct list
{
  void *element;
  list next;
};

list
cons(void *element, list l)
{
  ...
}

list
list_copy(list l)
{
  return l == NULL ? NULL : cons(l -> element, list_copy(l -> next));
}

struct queue
{
  list first;
  list last;
};

queue
queue_create(void)
{
  queue temp = malloc(sizeof(struct queue));
  temp -> last = temp -> first = cons(NULL, NULL);
  return temp;
}
```

```
void
queue_enq(queue q, void *element)
{
  q -> last -> next = cons(element, NULL);
  q -> last = q -> last -> next;
}

int
queue_empty(queue q)
{
  return q -> first -> next == NULL;
}

void *
queue_deq(queue q)
{
  void *temp;
  assert(!queue_empty(q));
  temp = q -> first -> next -> element;
  q -> first = q -> first -> next;
  return temp;
}

queue
queue_copy(queue q)
{
  queue temp = malloc(sizeof(struct queue));
  temp -> first = list_copy(q -> first);
  for(temp -> last = temp -> first;
      temp -> last -> next;
      temp -> last = temp -> last -> next)
    ;
  return temp;
}
```