Worksheet 18: Linked List Queue, pointer to Tail

```
struct link {
   TYPE value;
   struct link * next;
};

struct listQueue {
   struct slink *firstLink;
   struct slink *lastLink;
};

void ListQueueInit (struct listQueue *q) {
   struct slink *lnk = (struct slink *) malloc(sizeof(struct slink));
   assert(lnk != 0); /* lnk is the sentinel */
   lnk->next = 0;
   q->firstLink = q->lastLink = lnk;
}

void listQueueAddBack (struct listQueue *q, TYPE e) {
   struct slink *lnk = (struct slink *) malloc(sizeof(struct slink));
   assert(lnk != 0);
   lnk->value = e;
   lnk->next = 0;
   q->lastLink->next = lnk;
   q->lastLink = lnk;
}

TYPE listQueueFront (struct listQueue *q) {
   assert (! listQueueIsEmpty(q));
   return q->firstLink->next->value;
}


void listQueueRemoveFront (struct listQueue *q) {
    struct slink * lnk = q->firstLink->next;
    assert ( ! listQueueIsEmpty(q));
    q->firstLink->next = lnk->next;
    if(q->firstLink->next == 0)
        q->lastLink = q->firstLink;
    free (lnk);
}

int listQueueIsEmpty (struct listQueue *q) {
    return q->firstLink == q->lastLink;
}
```