Linked list with tail pointer

Typedef struct list\_type {

Node\_t \*head; (first pointer in the list)

Node\_t \*tail; (points to the end of the list)

} list\_t;

Head and tail are usually set to NULL

List\_add

Malloc new node\_t

Ensure next pointer is set to NULL

Ensure the next pointer

Next pointer of node\_t points to the new node\_t

Two cases

(list->head == NULL)

(list->head != NULL)

Void list\_add(list\_t \*list, void \*objPtr) {

Node\_t \*node = 9node\_t \*)malloc(sizeof(node\_t));

Node->dataPtr = objPtr;

Node->next = NULL;

If (Node->head == NULL) {

List->head = node;

}

Else {

List->tail->next = node;

}

List->tail = node;

}