

CSE108

LW 10

In this lab you will work on linked lists holding some texts.

The texts will be hold using the following structure:

text_t
int id int length char* text

Note that text is indicated with a pointer. Therefore, no storage is allocated for it in the structure and you should allocate its storage on the heap.

You will create 2 ordered linked lists and merge them.

PART 1 (2 Pts) Implement the following functions:

- **text_t getText(int id, char ch):** Creates and returns a text_t type variable having the given id. The text it holds will be a string having a random length in the range of 5 - 30 characters and will be obtained by repeating the giving character ch “length” times.
- **int printText(const text_t* text):** Prints the given text_t object. Returns the number of characters printed.

PART 2 (4 Pts) Implement the following functions:

- **node_t* createOrderedLL(unsigned int a, unsigned int b, unsigned int numOfElem, char ch):** Creates and returns a linked list such that:
 - n-th element of it will have an id of an+b,
 - it has “numOfElem” nodes,
 - the data in each node of it will be obtained by calling getText() with the calculated index and the given “ch”,
 - because id’s are monotone increasing the list is ordered with respect to the id’s.
- **int printLL(node_t* list):** Prints the linked list and returns the number of nodes it has.

PART 3 (3 Pts) Implement the following function:

- **node_t* mergeOrderedLists(node_t* list1, node_t* list2):** Merges the ordered lists without using extra space. (list1 and list2 will be destructed in this process)

PART4 (2 Pts) Write a function to free all dynamically allocated memory associated with a linked list. Note that a text string in a text_t type variable is also allocated on heap and should be freed.