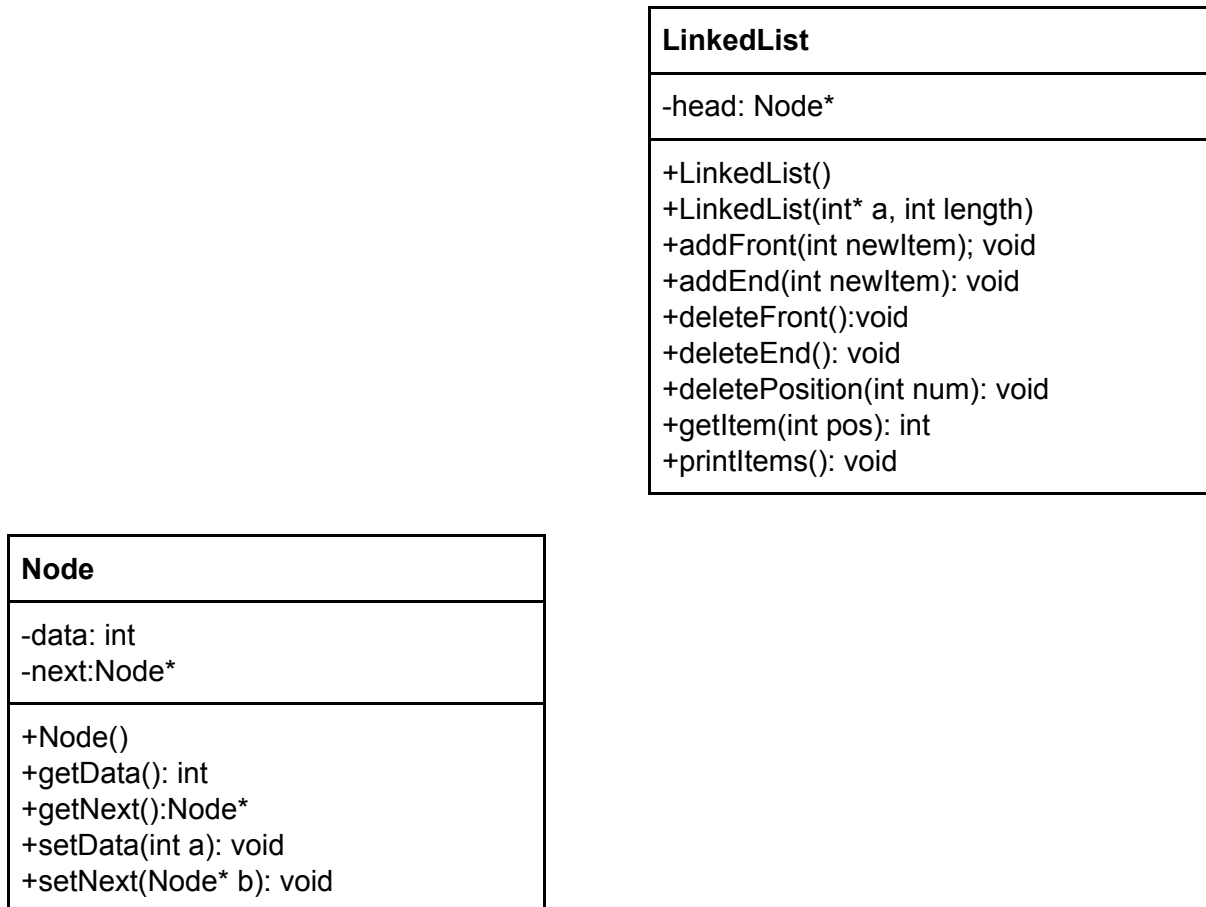


Prac 7

UML



Description

LinkedList

A constructor with no parameters, which makes an empty list.

A constructor that takes an array of integers and makes a linked list, containing all the elements of the array, in the same order. As the second parameter, it takes the size of the array.

void addFront(int newItem): The function inserts a new node, containing the newItem, at the beginning of the list

void addEnd(int newItem): The function inserts a new node, containing the newItem, at the end of the list.

void addAtPosition(int position, int newItem): The function inserts a new node, containing the newItem, such that it is the position-th member of the list. i.e. we assume the first element of the list is in position 1. If position is larger than the size of the list, the new item is added to the end of the list. If position < 1, the new item is added at the beginning of the list.

int search(int item): The function searched the list for the item, and if found, both prints the position of the item (followed by a space) and returns the position of the item in the list (positions start from 1). If not found, both prints 0 (followed by a space) and returns 0. Note that the returning type is different from what was explained in the search function in the lecture.

void deleteFront(): The function deletes the first element of the list.

void deleteEnd(): The function deletes the last element of the list.

void deletePosition(int position): The function deletes the element at the given position of the list. If the position is 1 or it is larger than the size of the list, only print "outside range".

int getItem(int position): The function both prints the value of the item (followed by a space) and returns the value of the item at the given position of the list. If beyond the size of the array, both prints std :: numeric_limits < int >:: max() 2 COMP SCI 2103 & 7103 Algorithm Design & Data Structure Semester 1, 2019 (followed by a space) and returns std :: numeric_limits < int >:: max().

void printItems(): The function prints the value of the items of the list from head to tail. In case of an empty list, it does not print anything

- A destructor that manually deletes all the elements that are still in the list.

Node

Node() is a default constructor of Node

getData() is a integer type getter which returns data

getNext() is a pointer type getter which returns pointers to Node

setData(int a) is a setter which sets data

setNext(Node* b) is a setter which sets pointers pointing to Node

Testing

Given input 1 27 9 19 AE 16
I expect output 1 27 9 19 16

Given input 49 83 13 56 DP 3
I expect output 49 83 56

Given input 3 47 81 9 GI 3 0
I expect output 81 3 47 81 9

Given input 98 56 72 33 AF 5
I expect output 5 98 56 72 33

Given input 32 45 9 88 S 45
I expect output 2 32 45 9 88