Linkedlist

#include <stdio.h>

#include <stdlib.h>

**typedef** **struct** node{

**int** data;

**struct** node \*next;

**struct** node \*prev;

}NODE;

**void** traverseForward(NODE \*head);

**void** traverseBackward(NODE \*tail);

//NODE \*head; prohibited

**int** main(**int** argc, **const** **char** \* argv[]) {

NODE \*head;

NODE \*tmp;

NODE \*tail;

traverseForward(head);

traverseBackward(tail);

**return** 0;

}

**void** addNode(NODE \*\* head){ // 1) headin degisme ihtimali varsa ve void dondureceksen pointerın pointerı kullanmalisin

}

NODE\* add2Node(NODE \* head){// 2)ya da tek pointer seklinde yollarsin sonra degiseni return head ile alirsin

**return** head;

}

/\*

3. yolda ise NODE \*head seklinde bir global degisken tanimlamak. cunku nereden degistirirsen degistir zaten degismis hali donecek. ama bu prohibited\*/

/\*void deleteNode{

}\*/

**void** traverseForward(NODE \*head){

NODE \*tmp=head;

**while** (tmp!=**NULL**) {

printf("%d ", tmp->data);

tmp=tmp->next;

}

printf("\n");

}

**void** traverseBackward(NODE \*tail){

NODE \*tmp=tail;

**while** (tmp!=**NULL**) {

printf("%d ", tmp->data);

tmp=tmp->prev;

}

printf("\n");

}