**DATA STRUCTURES LAB EXERCISES**

**EXPERIMENT-5**

#include <stdio.h>

#include <stdlib.h>

struct Node {

int data;

struct Node\* next;

};

struct Node\* createNode(int data) {

struct Node\* newNode = (struct Node\*)malloc(sizeof(struct Node));

newNode->data = data;

newNode->next = NULL;

return newNode;

}

void append(struct Node\*\* head, int data) {

struct Node\* newNode = createNode(data);

if (\*head == NULL) {

\*head = newNode;

return;

}

struct Node\* temp = \*head;

while (temp->next != NULL)

temp = temp->next;

temp->next = newNode;

}

struct Node\* mergeLists(struct Node\* list1, struct Node\* list2) {

if (list1 == NULL) return list2;

if (list2 == NULL) return list1;

struct Node\* temp = list1;

while (temp->next != NULL)

temp = temp->next;

temp->next = list2;

return list1;

}

void display(struct Node\* head) {

while (head != NULL) {

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

head = head->next;

}

printf("NULL\n");

}

int main() {

struct Node\* list1 = NULL;

struct Node\* list2 = NULL;

append(&list1, 1);

append(&list1, 3);

append(&list1, 5);

append(&list2, 2);

append(&list2, 4);

append(&list2, 6);

printf("First List: ");

display(list1);

printf("Second List: ");

display(list2);

struct Node\* mergedList = mergeLists(list1, list2);

printf("Merged List: ");

display(mergedList);

return 0;

}

