Create, insert and display Linked List

#include<stdio.h>

#include<stdlib.h>

struct Node {

int data;

struct Node \*link;

};

typedef struct Node node;

node \*new1, \*curr, \*start = NULL;

void create();

void display();

void insert();

void insert\_beg();

void insertAtPosition();

void insert\_end();

int main() {

int ch;

while(1) {

printf("1. Create 2. Insert 3. Display 4. Exit\n");

printf("Enter your choice: ");

scanf("%d", &ch);

switch(ch) {

case 1:

create();

break;

case 2:

insert();

break;

case 3:

display();

break;

case 4:

exit(0);

default:

printf("Wrong choice\n");

}

}

return 0;

}

void create() {

char ch;

do {

new1 = (node\*)malloc(sizeof(node));

printf("Enter value: \n");

scanf("%d", &new1->data);

if(start == NULL) {

start = new1;

curr = new1;

} else {

curr->link = new1;

curr = new1;

}

printf("Do you want to add another element? (Y/N)\n");

getchar();

scanf("%c", &ch);

} while(ch == 'y' || ch == 'Y');

curr->link = NULL;

}

void insert() {

int choice;

printf("1. Insert at First Position\n");

printf("2. Insert at Specific Position\n");

printf("3. Insert at End\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch(choice) {

case 1:

insert\_beg();

break;

case 2:

insertAtPosition();

break;

case 3:

insert\_end();

break;

default:

printf("Wrong choice\n");

}

}

void insert\_beg() {

new1 = (node\*)malloc(sizeof(node));

printf("Enter value to insert at the first position: ");

scanf("%d", &new1->data);

if(start==NULL){

start=new1;

new1->link=NULL;

return;

}

new1->link=start;

start=new1;

}

void insertAtPosition() {

int pos, i;

printf("Enter position where you want to insert the node: ");

scanf("%d", &pos);

if (pos < 0) {

printf("Invalid position\n");

return;

}

new1 = (node\*)malloc(sizeof(node));

printf("Enter value to insert: ");

scanf("%d", &new1->data);

if (pos == 0) {

new1->link = start;

start = new1;

return;

}

node \*temp = start;

for(i = 0; temp != NULL && i < pos - 1; i++) {

temp = temp->link;

}

if(temp == NULL) {

printf("Position out of range\n");

free(new1);

} else {

new1->link = temp->link;

temp->link = new1;

}

}

void insert\_end() {

new1 = (node\*)malloc(sizeof(node));

printf("Enter value to insert at the end: ");

scanf("%d", &new1->data);

if(start == NULL) {

start = new1;

new1->link=NULL;

return;

}

node \*temp = start;

while(temp->link != NULL) {

temp = temp->link;

}

new1->link=NULL;

temp->link = new1;

}

void display() {

printf("Ruqaiyya Mahreen 1BM23EE044\n");

node \*temp;

if(start == NULL) {

printf("Linked list is empty\n");

return;

}

printf("Elements of list: ");

temp = start;

while(temp != NULL) {

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

temp = temp->link;

}

printf("NULL\n");

}

