***Creation of Doubly Linked List***

#include <stdio.h>

#include <stdlib.h>

/\*

\* Basic structure of Node

\*/

struct node {

int data;

struct node \* prev;

struct node \* next;

}\*head, \*last;

/\*

\* Function used in this program

\*/

void createList(int n);

void displayListFromFirst();

void displayListFromEnd();

int main()

{

int n, choice;

head = NULL;

last = NULL;

printf("Enter the number of nodes you want to create: ");

scanf("%d", &n);

createList(n); // Create list of n nodes

printf("\nPress 1 to display list from First");

printf("\nPress 2 to display list from End : ");

scanf("%d", &choice);

if(choice==1)

{

displayListFromFirst();

}

else if(choice == 2)

{

displayListFromEnd();

}

return 0;

}

/\*\*

\* Create a doubly linked list of n nodes.

\* @n Number of nodes to be created

\*/

void createList(int n)

{

int i, data;

struct node \*newNode;

if(n >= 1)

{

head = (struct node \*)malloc(sizeof(struct node));

if(head != NULL)

{

printf("Enter data of 1 node: ");

scanf("%d", &data);

head->data = data;

head->prev = NULL;

head->next = NULL;

last = head;

/\*

\* Create rest of the n-1 nodes

\*/

for(i=2; i<=n; i++)

{

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

if(newNode != NULL)

{

printf("Enter data of %d node: ", i);

scanf("%d", &data);

newNode->data = data;

newNode->prev = last; // Link new node with the previous node

newNode->next = NULL;

last->next = newNode; // Link previous node with the new node

last = newNode; // Make new node as last/previous node

}

else

{

printf("Unable to allocate memory.");

break;

}

}

printf("\nDOUBLY LINKED LIST CREATED SUCCESSFULLY\n");

}

else

{

printf("Unable to allocate memory");

}

}

}

/\*\*

\* Displays the content of the list from beginning to end

\*/

void displayListFromFirst()

{

struct node \* temp;

int n = 1;

if(head == NULL)

{

printf("List is empty.");

}

else

{

temp = head;

printf("\n\nDATA IN THE LIST:\n");

while(temp != NULL)

{

printf("DATA of %d node = %d\n", n, temp->data);

n++;

/\* Move the current pointer to next node \*/

temp = temp->next;

}

}

}

/\*\*

\* Display the content of the list from last to first

\*/

void displayListFromEnd()

{

struct node \* temp;

int n = 0;

if(last == NULL)

{

printf("List is empty.");

}

else

{

temp = last;

printf("\n\nDATA IN THE LIST:\n");

while(temp != NULL)

{

printf("DATA of last-%d node = %d\n", n, temp->data);

n++;

/\* Move the current pointer to previous node \*/

temp = temp->prev;

}

}

}

***OUTPUT***

Enter the number of nodes you want to create: 5

Enter data of 1 node: 32

Enter data of 2 node: 64

Enter data of 3 node: 85

Enter data of 4 node: 25

Enter data of 5 node: 16

DOUBLY LINKED LIST CREATED SUCCESSFULLY

Press 1 to display list from First

Press 2 to display list from End : 1

DATA IN THE LIST:

DATA of 1 node = 32

DATA of 2 node = 64

DATA of 3 node = 85

DATA of 4 node = 25

DATA of 5 node = 16