

Stack implementation using linked list

```
#include <stdio.h>
#include <stdlib.h>

struct node {
    int data;
    struct node * next;
};

struct node * head;

void push (int value) {
    struct node * newnode;
    newnode = (struct node *) malloc (sizeof (struct node));
    newnode->data = value;
    newnode->next = 0;
    if (head == 0) {
        head = newnode;
    }
    else {
        newnode->next = head;
        head = newnode;
    }
}

void pop() {
    struct node * temp;
    temp = head;
    if (head == 0) {
        printf ("Stack is empty");
    }
    else {
        head = head->next;
        free (temp);
    }
}
```

void display () {

struct node * temp;

temp = head;

while (temp != NULL) {

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

printf ("\n");

temp = temp->next; }

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

}

int main () {

push (5);

push (10);

push (15);

push (20);

display ();

pop ();

display ();

return 0;

}

Output →

20

15

10

5

15

10

5

Queue implementation using linked list :-

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct node {
```

```
    int data;
```

```
    struct node * next;
```

```
};
```

```
struct node * head;
```

```
void enqueue (int value) {
```

```
    struct node * newnode, * temp;
```

```
    newnode = (struct node *) malloc (sizeof  
                                         struct node);
```

```
    newnode->data = value;
```

```
    newnode->next = NULL;
```

```
    if (head == NULL) {
```

```
        head = newnode;
```

```
        temp = head;
```

```
    }
```

```
    else {
```

```
        temp->next = newnode;
```

```
        temp = newnode;
```

```
    }
```

```
}
```

```
void dequeue () {
```

```
    struct node * temp;
```

```
    temp = head;
```

```
    head = head->next;
```

```
    free (temp);
```

```
}
```

```

void display() {
    struct node *temp = head;
    while (temp != NULL) {
        printf("%d", temp->data);
        temp = temp->next;
    }
    printf("\n");
}

```

```

int main() {

```

```

    enqueue(5);

```

```

    enqueue(10);

```

```

    enqueue(15);

```

```

    enqueue(20);

```

```

    display();

```

```

    deque();

```

```

    display();

```

```

    deque();

```

```

    display();
}

```

Output →

5 10 15 20

10 15 20 // First deque

15 20 // Second deque

6 (B3)

Write code maximum sum
of linked list

Output :

Case 1:

Input :

head = [5, 4, 2, 1]

Output = 6

Expected = 6

Case 2:

Input

head = [4, 2, 2, 3]

Output = 7

Expected = 7

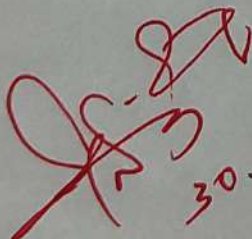
Case 3:

Input:

head = [1, 10000]

Output = 10000

Expected = 10000


30.01.24