**CO1**

**Stack**

**Code :**

#include<stdio.h>

#define n 5

int s[n],top=-1;

void push();

void pop();

void Top();

void display();

void main() {

int o,c=1;

while(c==1) {

printf("Enter any of the below option number\n1.push\n2.pop\n3.top\n4.display\n");

scanf("%d",&o);

switch(o) {

case 1 : push();

break;

case 2 : pop();

break;

case 3 : Top();

break;

case 4 : display();

break;

}

printf("Do you want to continue(0/1)\n");

scanf("%d",&c);

}

}

void push() {

int x;

printf("Enter an element to push\n");

scanf("%d",&x);

if(top==n-1) {

printf("\nOverflow\n");

} else {

top++;

s[top]=x;

}

}

void pop() {

if(top==-1) {

printf("\nUnderflow\n");

} else {

printf("Popped element is %d",s[top]);

top--;

}

}

void Top() {

if(top==-1) {

printf("\nUnderflow\n");

} else {

printf("Top element is %d",s[top]);

}

}

void display() {

if(top==-1) {

printf("\nUnderflow\n");

} else {

printf("Stack elements are\n");

for (int i=top; i>=0; i--) {

printf("%d\n",s[i]);

}

}

}

Text

Description automatically generated

**Queue**

**Code :**

#include<stdio.h>

#define n 5

int q[n],front=-1,rear=-1;

void insert();

void delete();

void peak();

void display();

void main() {

int o,c=1;

while(c==1) {

printf("Enter any of the below option number\n1.Insert\n2.Delete\n3.Peak\n4.Display\n");

scanf("%d",&o);

switch(o) {

case 1 : insert();

break;

case 2 : delete();

break;

case 3 : peak();

break;

case 4 : display();

break;

default : printf("Invalid entry");

}

printf("Do you want to continue(0/1)\n");

scanf("%d",&c);

}

}

void insert() {

int x;

printf("Enter an element to insert\n");

scanf("%d",&x);

if(rear==n-1) {

printf("\nOverflow\n");

//return;

} else if(front==-1 && rear==-1) {

front=rear=0;

} else {

rear++;

}

q[rear]=x;

}

void delete() {

if(front==-1 || front>rear) {

printf("\nUnderflow\n");

} else {

printf("Deleted element is %d",q[front]);

front++;

if(front>rear) {

front=rear=-1;

}

}

}

void peak() {

if(front==-1 || front>rear) {

printf("\nUnderflow\n");

} else {

printf("Peak element is %d",q[front]);

}

}

void display() {

if(front==-1 || front>rear) {

printf("\nUnderflow\n");

} else {

printf("Queue elements are\n");

for (int i=front;i<=rear;i++) {

printf("%d\n",q[i]);

}

}}

Text

Description automatically generated

**Linked List**

**Code :**

#include<stdio.h>

#include<stdlib.h>

struct node {

int data;

struct node \*next;};

struct node \*head,\*newnode,\*temp,\*prev;

void InsertBeg() {

int m;

printf("Enter value to insert");

scanf("%d",&m);

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

newnode->data=m;

newnode->next=head;

head=newnode;

}

void InsertBetween() {

int m,x;

printf("Enter value to insert");

scanf("%d",&m);

printf("Enter after which value to insert");

scanf("%d",&x);

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

newnode->data=m;

temp=head;

while(temp->next!=0) {

if(temp->data==x) {

break;}

temp=temp->next;}

newnode->next=temp->next;

temp->next=newnode;}

void InsertEnd() {

int m;

printf("Enter value to insert");

scanf("%d",&m);

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

newnode->data=m;

while(temp->next!=0) {

temp=temp->next;}

temp->next=newnode;

newnode->next=0;}

void DeleteBeg() { head=head->next;}

void DeleteBetween() {

int x;

printf("Enter the node data for position");

scanf("%d",&x);

temp=head;

while(temp->data!=x) {

temp=temp->next;}

temp->next=temp->next->next;}

void DeleteEnd() {

temp=head;

while(temp->next!=0) {

prev=temp;

temp=temp->next;}

prev->next=NULL;}

void display() {

temp=head;

while(temp!=0) {

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

temp=temp->next;}}

void addnode() {

int o=1,c=0,m;

printf("Enter value to insert");

scanf("%d",&m);

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

newnode->data=m;

newnode->next=0;

if(head==0) {

head=temp=newnode;

} else {

temp->next=newnode;

temp=newnode;}}

void main() {

int c=1,o,i,m;

head=0;

while(c==1) {

printf("Enter any of the below option number\n1.AddNode\n2.Insert-Begining\n3.Insert-End\n4.Insert-Between\n5.Delete-Begining\n6.Delete-End\n7.Delete-Between\n8.Dispaly\n");

scanf("%d",&o);

switch(o) {

case 1 : addnode();

break;

case 2 : InsertBeg();

break;

case 3 : InsertBetween();

break;

case 4 : InsertEnd();

break;

case 5 : DeleteEnd();

break;

case 6 : DeleteEnd();

break;

case 7 : DeleteEnd();

break;

case 8 : display();

break;

}

printf("Do you want to continue(0/1)\n");

scanf("%d",&c);

}

}

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

**Doubly Linked List**

**Code :**

#include<stdio.h>

#include<stdlib.h>

struct node {

struct node \*prev;

int data;

struct node \*next;};

struct node \*head,\*newnode,\*temp,\*ptr;

void Beg() {

newnode->prev = head;

head=newnode;

newnode->next = head->prev;}

void Between() {

int x;

temp=head;

printf("Enter the node data for position");

scanf("%d",&x);

while(temp->data!=x) {

temp=temp->next;}

newnode->next=temp->next;

temp->next=newnode;

temp->next->prev=newnode;}

void End() {

temp=head;

while(temp->next!=NULL) {

temp=temp->next;}

temp->next=newnode;

newnode->prev=temp;}

void DelBeg() {head=head->next;}

void DelBetween() {

int x;

temp=head;

printf("Enter the node data for position");

scanf("%d",&x);

while(temp->data!=x) {

temp=temp->next;}

printf("asa");

ptr=temp;

temp->next->next->prev=ptr;

temp->next=ptr->next->next;}

void DelEnd() {

temp=head;

while(temp->next!=NULL) {

temp=temp->next;}

temp->next=newnode;

newnode->prev=temp;}

void display() {

temp=head;

int count=0;

while(temp!=0) {

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

count++;

temp=temp->next;}}

void addnode() {

int o=1,c=0;

head=0;

while(o==1) {

if(head==0) {

head=temp=newnode;

} else {

temp->next=newnode;

newnode->prev=temp;

temp=newnode;}

printf("Do you want to continue insertion (0/1) ");

scanf("%d",&o);}}

void CreateNode() {

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

printf("Enter the data ");

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

newnode->next=NULL;

newnode->prev=NULL;}

void main() {

int c=1,o,i,m;

while(c==1) {

printf("Enter any of the below option number\n1.AddNode\n2.Insert-Begining\n3.Insert-End\n4.Insert-Between\n5.Delete-Begining\n6.Delete-End\n7.Delete-Between\n8.Dispaly\n");

scanf("%d",&o);

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

switch(o) {

case 1 : CreateNode();

addnode();

break;

case 2 : CreateNode();

Beg(m);

break;

case 3 : CreateNode();

End(m);

break;

case 4 : CreateNode();

Between();

break;

case 5 : DelBeg();

break;

case 6 : DelEnd();

break;

case 7 : DelBetween();

break;

case 8 : display();

break;

}

printf("Do you want to continue(0/1)\n");

scanf("%d",&c);

}

}