8)

Disha-B
1BM19CS050

```
void push(struct node **top, int d)
{
    struct node *temp, n;
    temp = (struct node*) malloc (sizeof(struct node));
    if( temp == NULL)
    {
        printf("Stack is full\n");
    }
    temp -> data = d;
    temp -> next = *top;
    *top = temp;
    printf("%d is pushed\n", d);
}
void pop(struct node** top)
{
    struct node* temp;
    if(*top == NULL)
    {
        printf(" stack underflow\n");
        return;
    }
    temp = *top;
    printf("%d popped\n", temp->data);
    *top = (*top)->next;
    free(temp);
}
void display(struct node* top)
{
    if(top == NULL)
    {
        printf("No elements present in stack\n");
```

```
return;
}
while(top!=NULL)
{ printf("%d", top->data);
  top = top->next;
}
printf("\n");
}
void insert(int d)
{ struct node *n;
  n = (struct node *)malloc(sizeof(struct node));
  if(n==NULL)
  {
    printf("Queue Overflow\n");
    return;
  }
  n->data=d;
  if(front==NULL)
  {
    front=n;
    rear=n;
    front->next=NULL;
    rear->next=NULL;
  }
  else
  {
    rear->next=n;
    rear=n;
    rear->next=NULL;
  }
```

Disha.B
1BM19CS050

```
    printf("%d is inserted\n", d);
}
void delete()
{
  struct node* temp;
  if(front == NULL)
  {
    printf("Queue underflow\n");
    return;
  }
  temp = front;
  printf("%d deleted\n", temp->data);
  front = front->next;
  free(temp);
}
void display_queue()
{
  struct node *temp;
  temp = front;
  if(front == NULL)
  {
    printf("\n Empty queue\n");
  }
  else
  {
    printf("\n Queue elements:\n");
    while(temp != NULL)
    {
      printf("%d", temp->data);
      temp = temp->next;
    }
```


Disha

```
    printf("\n");
    }
}
int main()
{
struct node *stack = NULL;
printf("STACK OPERATIONS\n");
printf("1.Push \t2.Pop \t3.Display\t4.Exit\n");
int choice, item;
printf("Enter your choice:");
scanf("%d", &choice);
while(choice != 4)
 { switch(choice)
   {
     case 1: printf("Enter data to be pushed:");
             scanf("%d", &item);
             push(&stack, item);
             break;
     case 2: pop(&stack);
             break;
     case 3: display(stack);
             break;
   }
   printf("1.Push\t2.Pop\t3.Display\t4.Exit\n");
   printf("Enter your choice:");
   scanf("%d", &choice);
 }
printf("End of stack operations\n\n");
printf("QUEUE OPERATIONS\n");
printf("1.Insert\t2.Delete\t3.Display\t4.Exit\n");
```

Disha · B
1BM19CS050

```
printf("Enter your choice:");
scanf("%d", &choice);
while(choice != 4)
{ switch(choice)
  { case1: printf("Enter the data to be inserted:");
           scanf("%d", &item);
           insert(item);
           break;
    case2: delete();
           break;
    case3: display_queue();
           break;
  }
  printf("1.Push\t2. Pop\t3. Display\t4. Exit\n");
  printf("Enter your choice:");
  scanf("%d", &choice);
}
printf("End of queue operations\n");
return 0;
}
```