LAB PROGRA-3

```
# include < stdio.h >
# define MAX 50
void insert();
void delete();
void display();
int queue_array[MAX];
int rear = -1;
int fron = -1;
main ()
{
int choice;
while(1)
{
printf(" 1. Insert element to queue \n");
printf(" 2. Delete element from queue \n");
printf(" 3. Display all elements of queue \n");
printf(" 4. Quit \n");
printf(" Enter your choice: ");
scanf(" %d", & choice);
switch(choice)
{
case 1 :  insert();
          break;
case 2: delete();
        break;
case 3: display();
         break;
case 4 : exit(1);
          ~~default~~
deafult: printf("Wrong choice\n");
 }
}
}
void insert()
{
 int add_item;
 if(rear == MAX-1)
   printf("Queue Overflow\n");
 else
 {
  if (front == -1)
     front = 0;
     printf("Insert the element in queue:");
     scanf("%d", &add_item);
    rear = rear+1;
    queue_array[rear] = add_item;
 }
}
```

```
void delete()
{
 if (front == -1 || front > rear)
{
 printf("Queue Underflow \n");
          return;
}
else
 {
 printf("Element deleted from queue is: %d
queue_array[front]);
        front = fron + 1
 }
}
void display()
{
 int i;
 if (front == -1)
      printf("Queue is empty \n");
 else
 {
  printf ("Queue is : \n");
for (i= front ; i<= rear ; i++)
    printf ("%d", queue-array[i]);
     printf ("\n");
}
 }
```

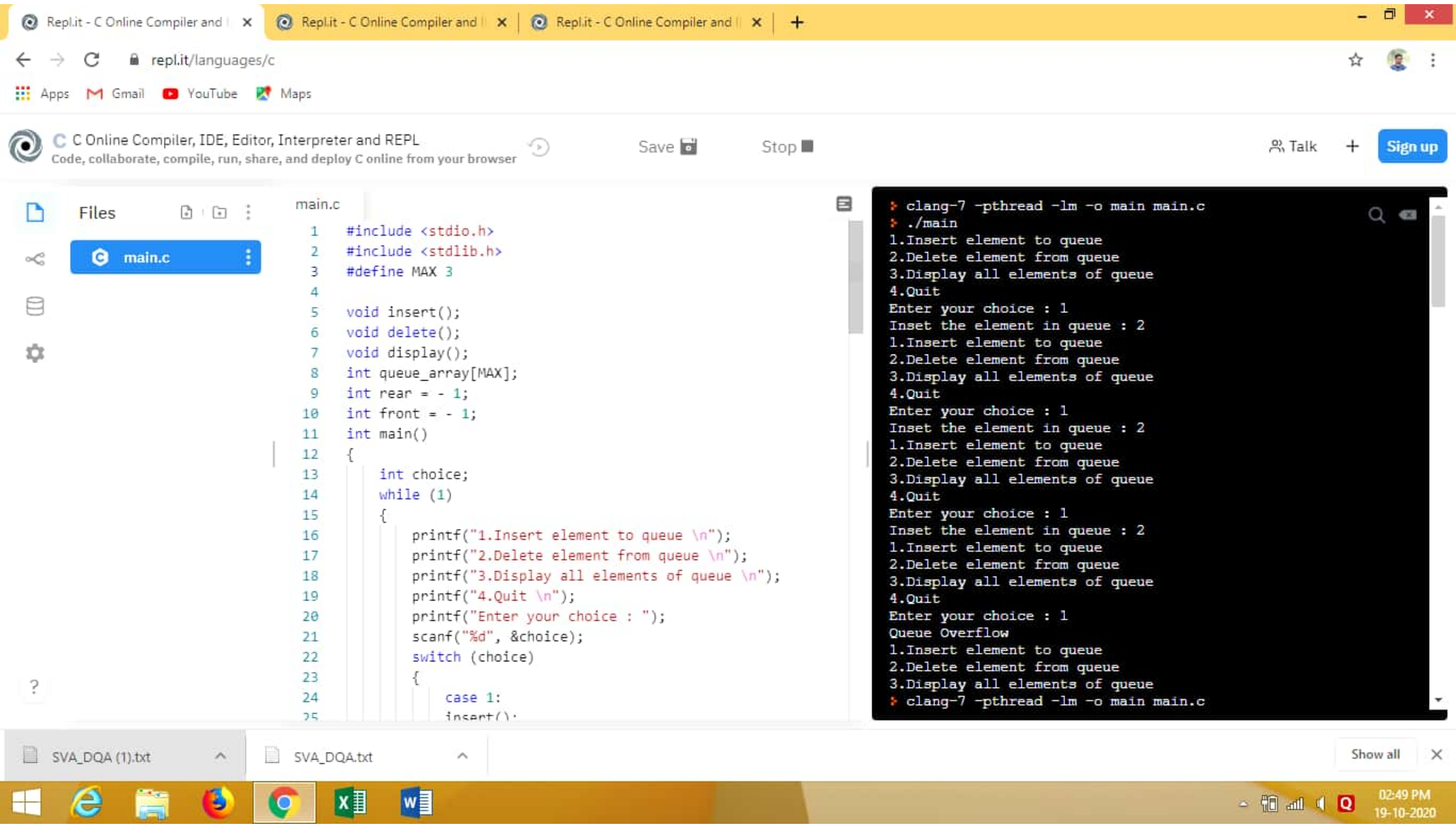
LAB PROGAM-4

```
# include <stdio.h>
# include <stdlib.h>
# define SIZE 6
int a[SIZE], t;
 int front = -1;
  int rear = -1;

int isEmpty()
{
if (rear==-1 && front==-1)
      return 1;
   else
      return 0;
 }

int isfull()
 {
  if (front == (rear+1)% SIZE)
   return 1;
   else
   return 0;
   }
```

```
void Enqueue ( int x )
{
 if (isFull()
   printf ("The queue is full \n");
 else if (isEmpty ())
  {
  front = 0;
   rear = 0;
   a[rear] = x;
 }
 else
  {
  rear = (rear + 1) % SIZE;
      a[rear] = x ;
   }
  }


int Dequeue()
{
 int x ;
  if (isEmpty())
   printf (" The queue is empty. \n");
   else if (front = = rear)
   {
     x = a[front];
```

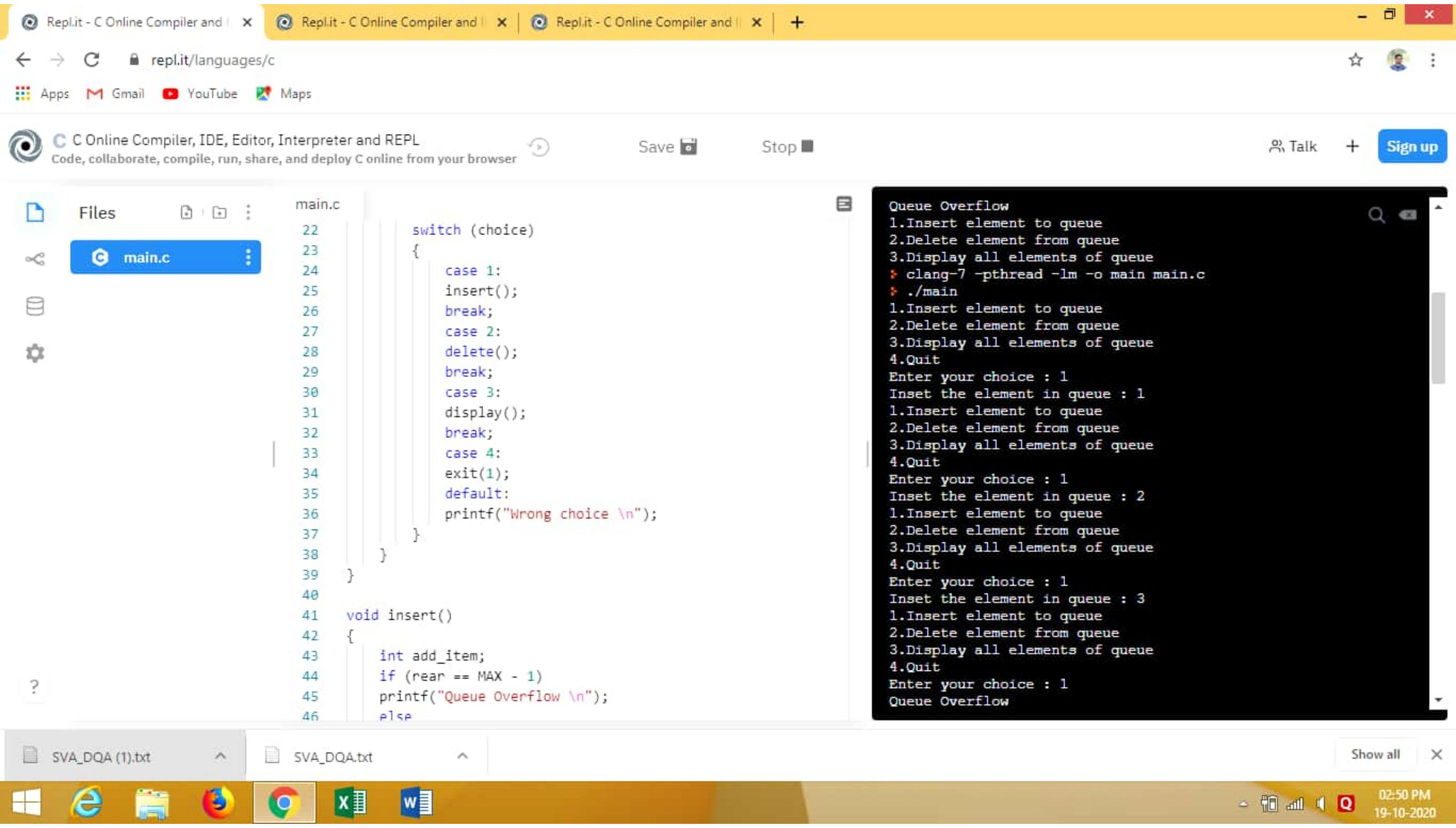
```
        front = -1;
        rear = -1;
      printf(" The element  was removed \n");
}
else
{
  x = a[front];
  front = (front+1) % SIZE;
  printf("The element was removed \n");
}
return x;
}
void display()
{
  if (front == -1)
   {
     printf("\nQueue is Empty");
         return;
   }
printf("\nElement in  Circular Queue are : \n
 if ( rear >= front)
  {
    for (int i = front; i<= rear; i++)
      printf("%d\n", a[i]);
    }
```

```
else
 {
 for (int i= front; i< SIZE ; i++)
      printf("%d \n", a[i]);

    for (int i=0; i<=rear; i++)
      printf("%d \n", a[i]);
  }
 }
int main()
{
 int n, a;
  while (1)
 {
 printf ("Enter the operation \n 1- Inser\n
2- delete \n 3- Display \n 4-Exit \n");
 scanf("%d", &n);
 switch(n)
  {
 case 1: printf("Enter the element \n");
         scanf ("%d ", &a);
         Enqueue (a);
          break;
case 2: Dequeue();
        break;

case 3: Display();
        break;

case 4: exit(0);

deafult: printf("There is no such operation \n");
 }
}
return 0;
```


Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:49 PM
19-10-2020

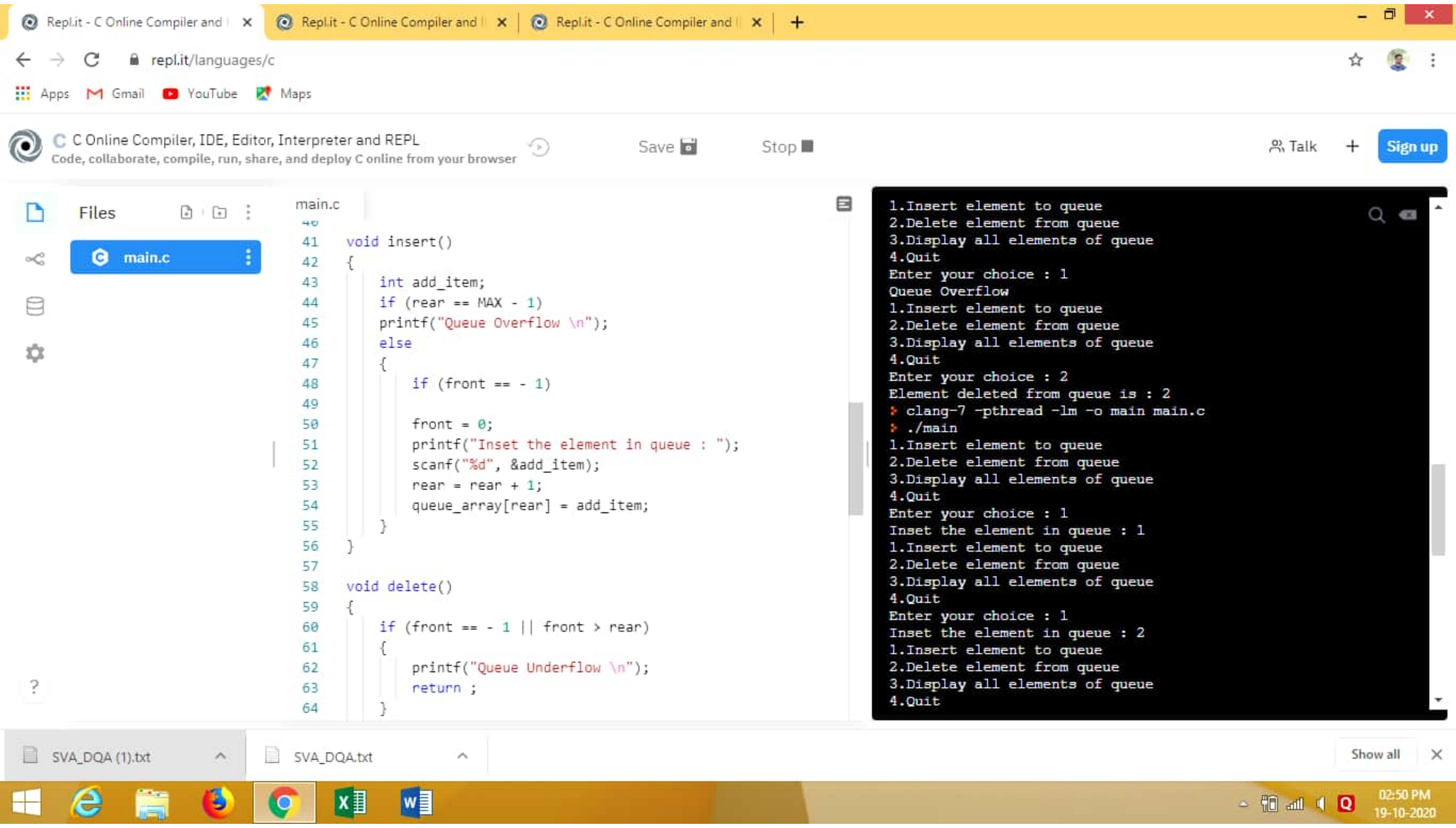
```c
#include <stdio.h>
#include <stdlib.h>
#define MAX 3

void insert();
void delete();
void display();
int queue_array[MAX];
int rear = - 1;
int front = - 1;
int main()
{
    int choice;
    while (1)
    {
        printf("1.Insert element to queue \n");
        printf("2.Delete element from queue \n");
        printf("3.Display all elements of queue \n");
        printf("4.Quit \n");
        printf("Enter your choice : ");
        scanf("%d", &choice);
        switch (choice)
        {
            case 1:
            insert();
```

```
› clang-7 -pthread -lm -o main main.c
› ./main
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 2
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 2
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 2
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Queue Overflow
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
› clang-7 -pthread -lm -o main main.c
```


Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:50 PM
19-10-2020

```
        switch (choice)
        {
            case 1:
            insert();
            break;
            case 2:
            delete();
            break;
            case 3:
            display();
            break;
            case 4:
            exit(1);
            default:
            printf("Wrong choice \n");
        }
    }
}

void insert()
{
    int add_item;
    if (rear == MAX - 1)
    printf("Queue Overflow \n");
    else
```

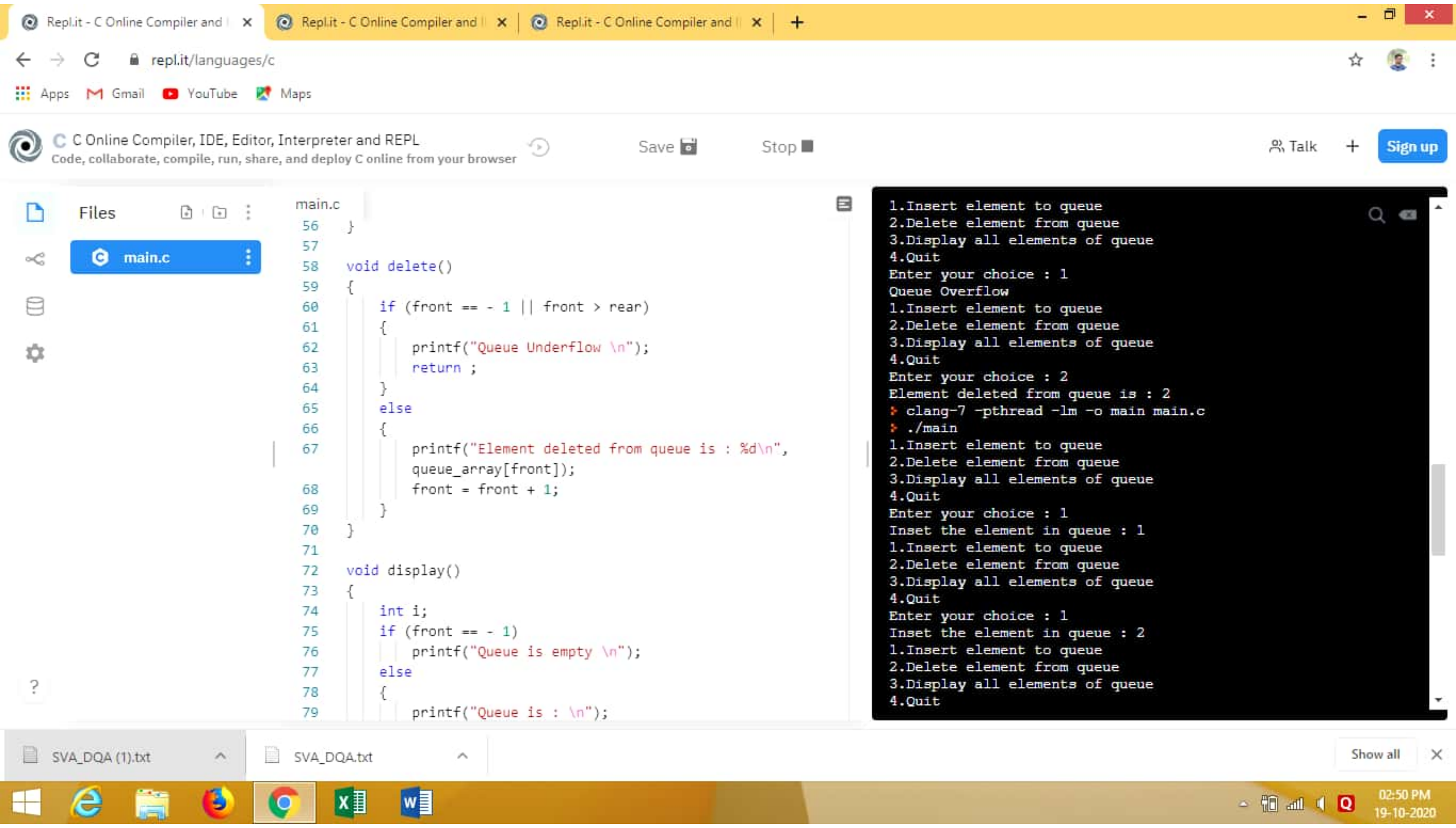
```
Queue Overflow
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
› clang-7 -pthread -lm -o main main.c
› ./main
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 1
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 2
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 3
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Queue Overflow
```



Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:50 PM
19-10-2020

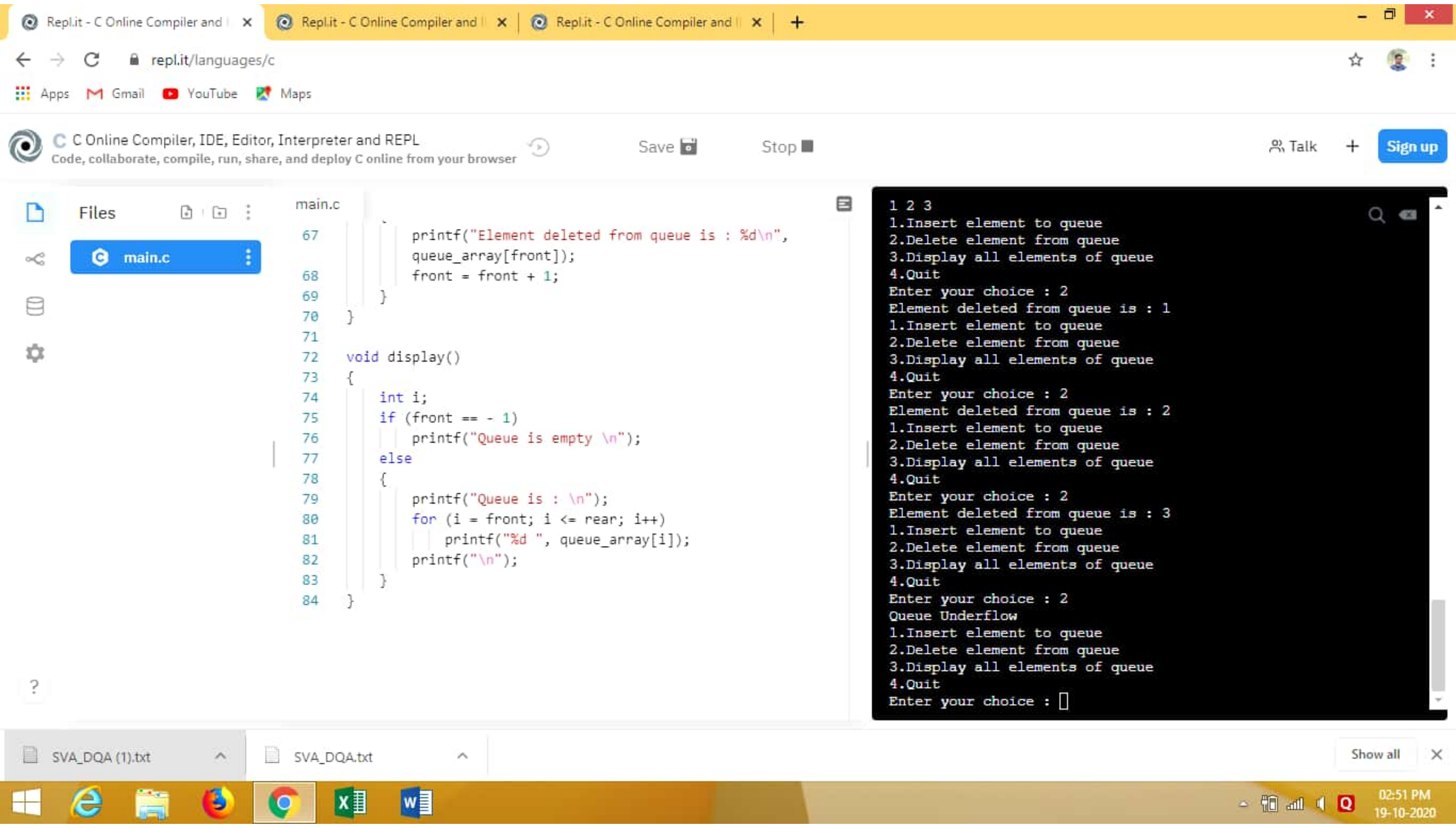
```c
void insert()
{
    int add_item;
    if (rear == MAX - 1)
    printf("Queue Overflow \n");
    else
    {
        if (front == - 1)

        front = 0;
        printf("Inset the element in queue : ");
        scanf("%d", &add_item);
        rear = rear + 1;
        queue_array[rear] = add_item;
    }
}

void delete()
{
    if (front == - 1 || front > rear)
    {
        printf("Queue Underflow \n");
        return ;
    }
```

```
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Queue Overflow
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 2
Element deleted from queue is : 2
> clang-7 -pthread -lm -o main main.c
> ./main
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 1
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 2
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
```


Repl.it - C Online Compiler and
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:50 PM
19-10-2020

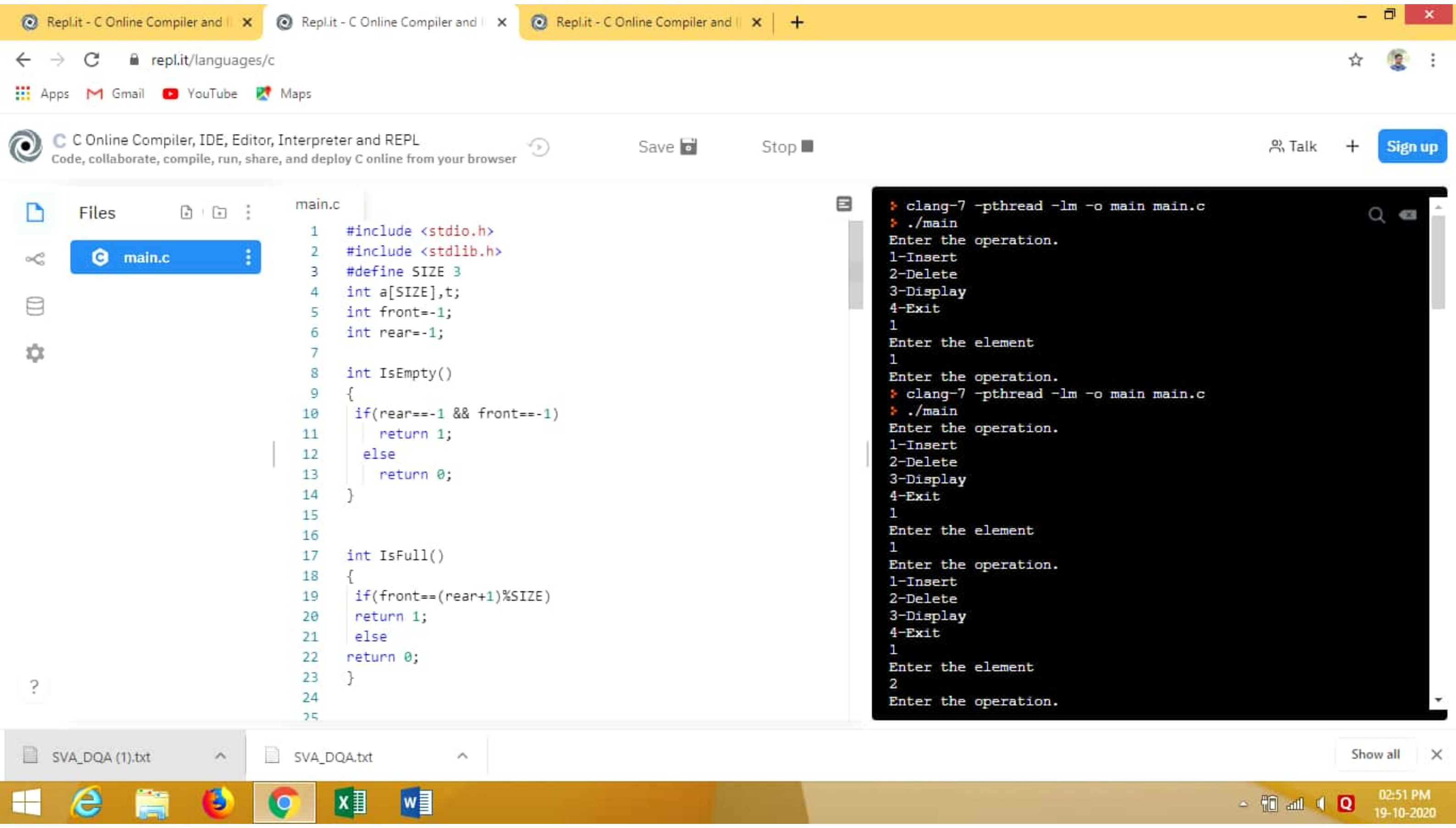
```c
}

void delete()
{
    if (front == - 1 || front > rear)
    {
        printf("Queue Underflow \n");
        return ;
    }
    else
    {
        printf("Element deleted from queue is : %d\n", queue_array[front]);
        front = front + 1;
    }
}

void display()
{
    int i;
    if (front == - 1)
        printf("Queue is empty \n");
    else
    {
        printf("Queue is : \n");
```

```
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Queue Overflow
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 2
Element deleted from queue is : 2
› clang-7 -pthread -lm -o main main.c
› ./main
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 1
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 2
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
```


Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:51 PM
19-10-2020

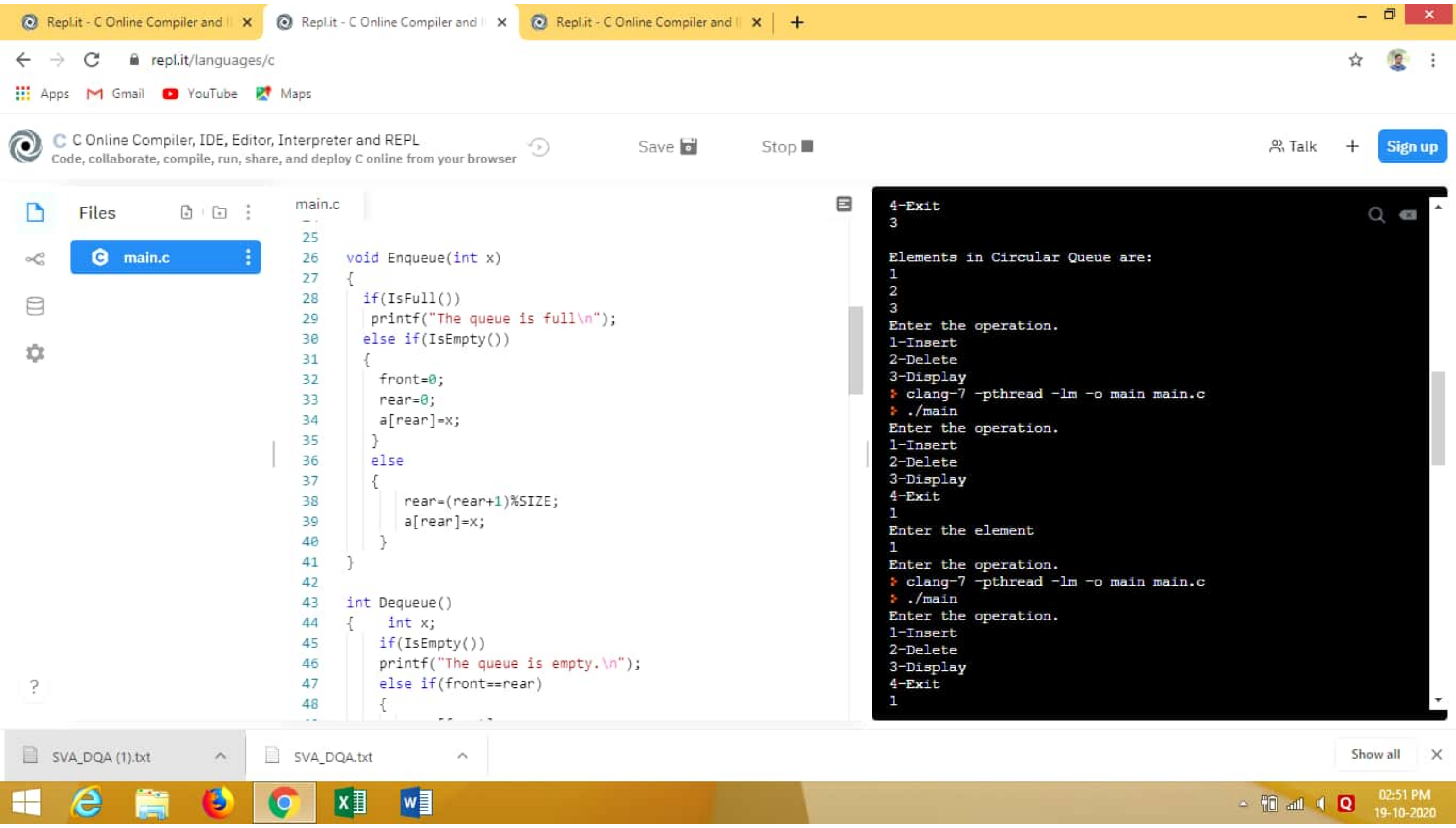
```c
        printf("Element deleted from queue is : %d\n",
        queue_array[front]);
        front = front + 1;
    }
}

void display()
{
    int i;
    if (front == - 1)
        printf("Queue is empty \n");
    else
    {
        printf("Queue is : \n");
        for (i = front; i <= rear; i++)
            printf("%d ", queue_array[i]);
        printf("\n");
    }
}
```

```
1 2 3
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 2
Element deleted from queue is : 1
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 2
Element deleted from queue is : 2
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 2
Element deleted from queue is : 3
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 2
Queue Underflow
1.Insert element to queue
2.Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 
```


Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:51 PM
19-10-2020

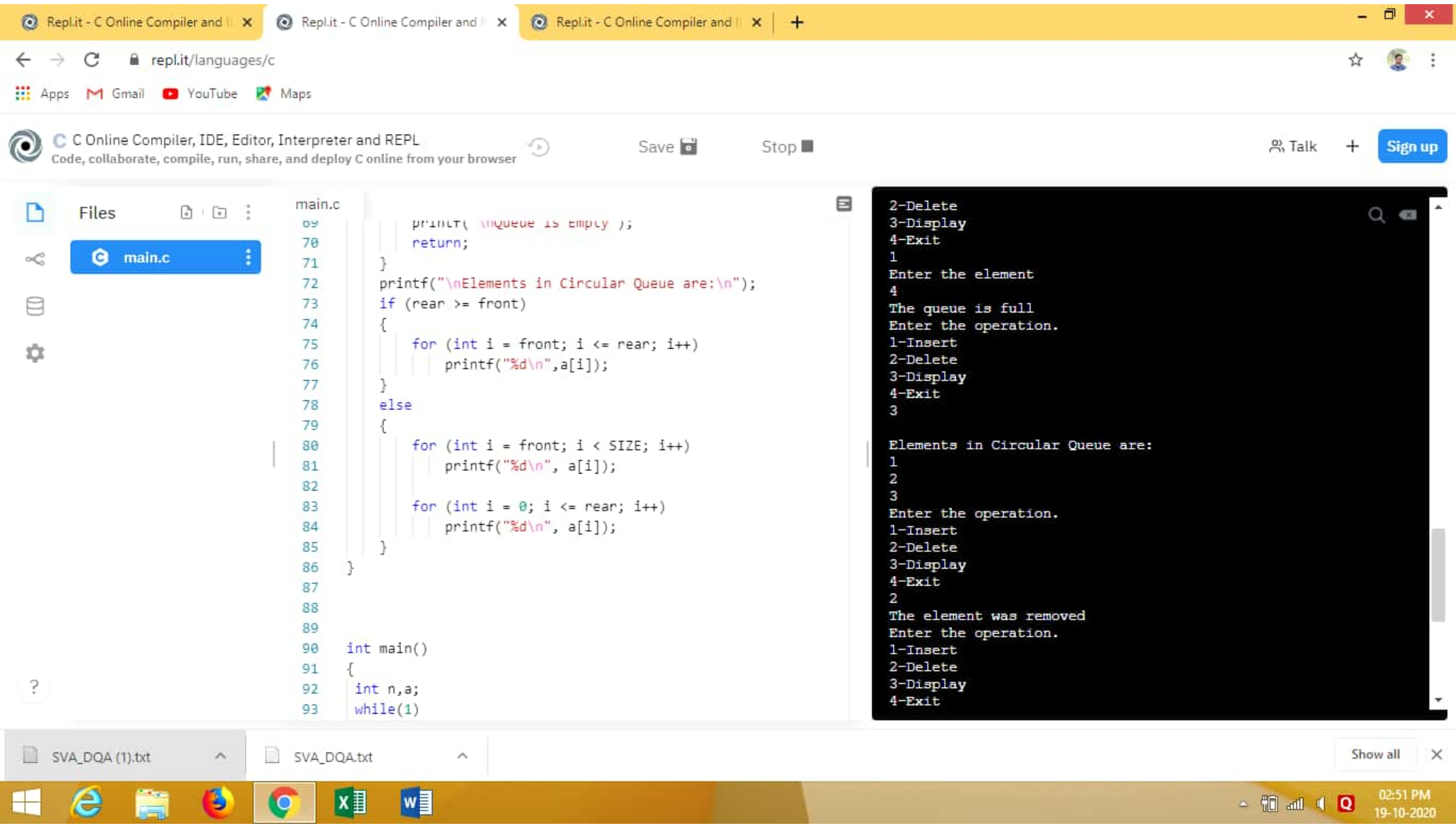
```
#include <stdio.h>
#include <stdlib.h>
#define SIZE 3
int a[SIZE],t;
int front=-1;
int rear=-1;

int IsEmpty()
{
 if(rear==-1 && front==-1)
   return 1;
  else
   return 0;
}


int IsFull()
{
 if(front==(rear+1)%SIZE)
 return 1;
 else
return 0;
}

```

```
> clang-7 -pthread -lm -o main main.c
> ./main
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
1
Enter the element
1
Enter the operation.
> clang-7 -pthread -lm -o main main.c
> ./main
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
1
Enter the element
1
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
1
Enter the element
2
Enter the operation.
```


Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:51 PM
19-10-2020

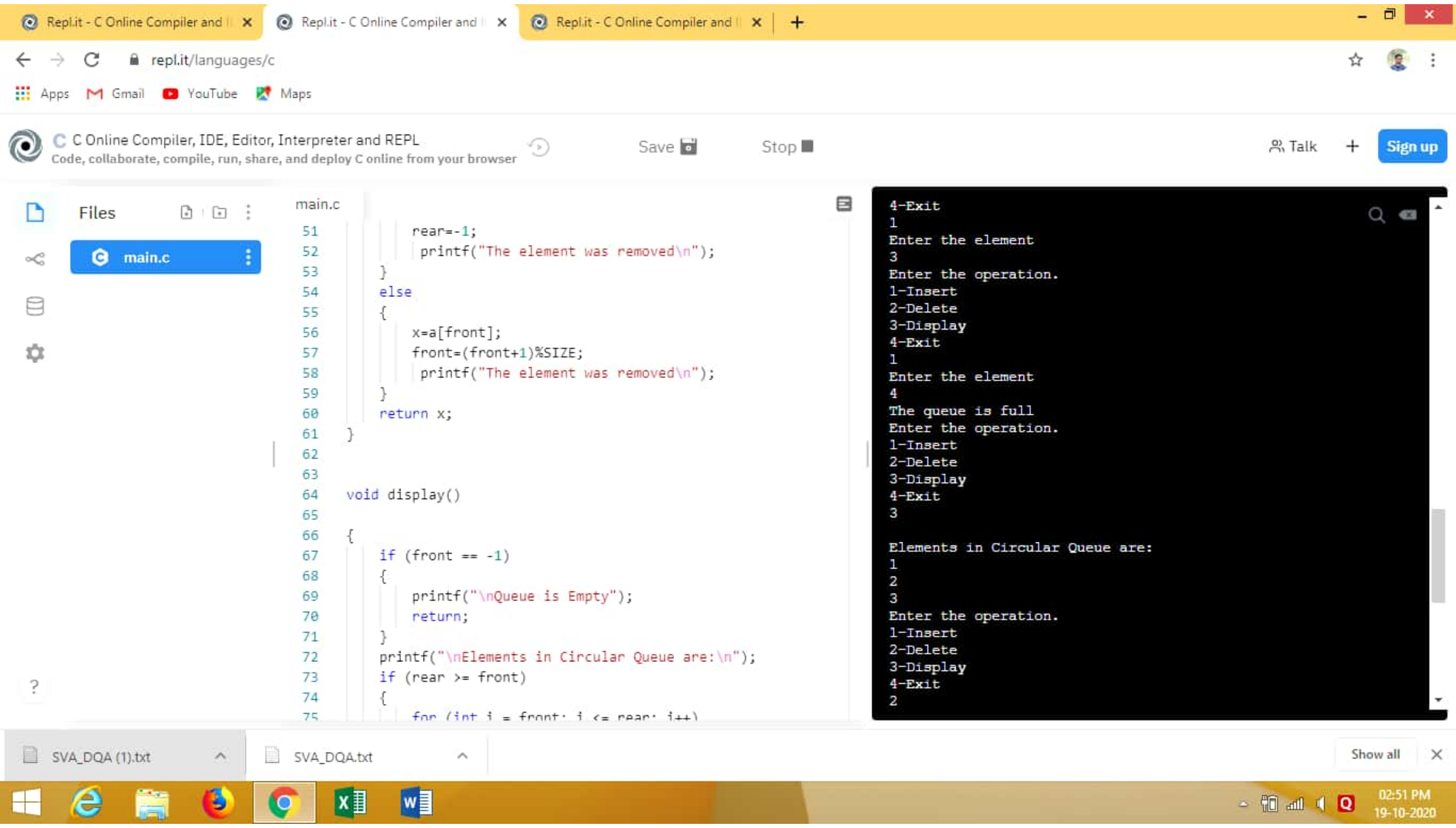
```c
void Enqueue(int x)
{
  if(IsFull())
   printf("The queue is full\n");
  else if(IsEmpty())
  {
    front=0;
    rear=0;
    a[rear]=x;
  }
   else
   {
      rear=(rear+1)%SIZE;
      a[rear]=x;
   }
}

int Dequeue()
{    int x;
    if(IsEmpty())
    printf("The queue is empty.\n");
    else if(front==rear)
    {
```

```
4-Exit
3

Elements in Circular Queue are:
1
2
3
Enter the operation.
1-Insert
2-Delete
3-Display
› clang-7 -pthread -lm -o main main.c
› ./main
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
1
Enter the element
1
Enter the operation.
› clang-7 -pthread -lm -o main main.c
› ./main
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
1
```


Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:51 PM
19-10-2020

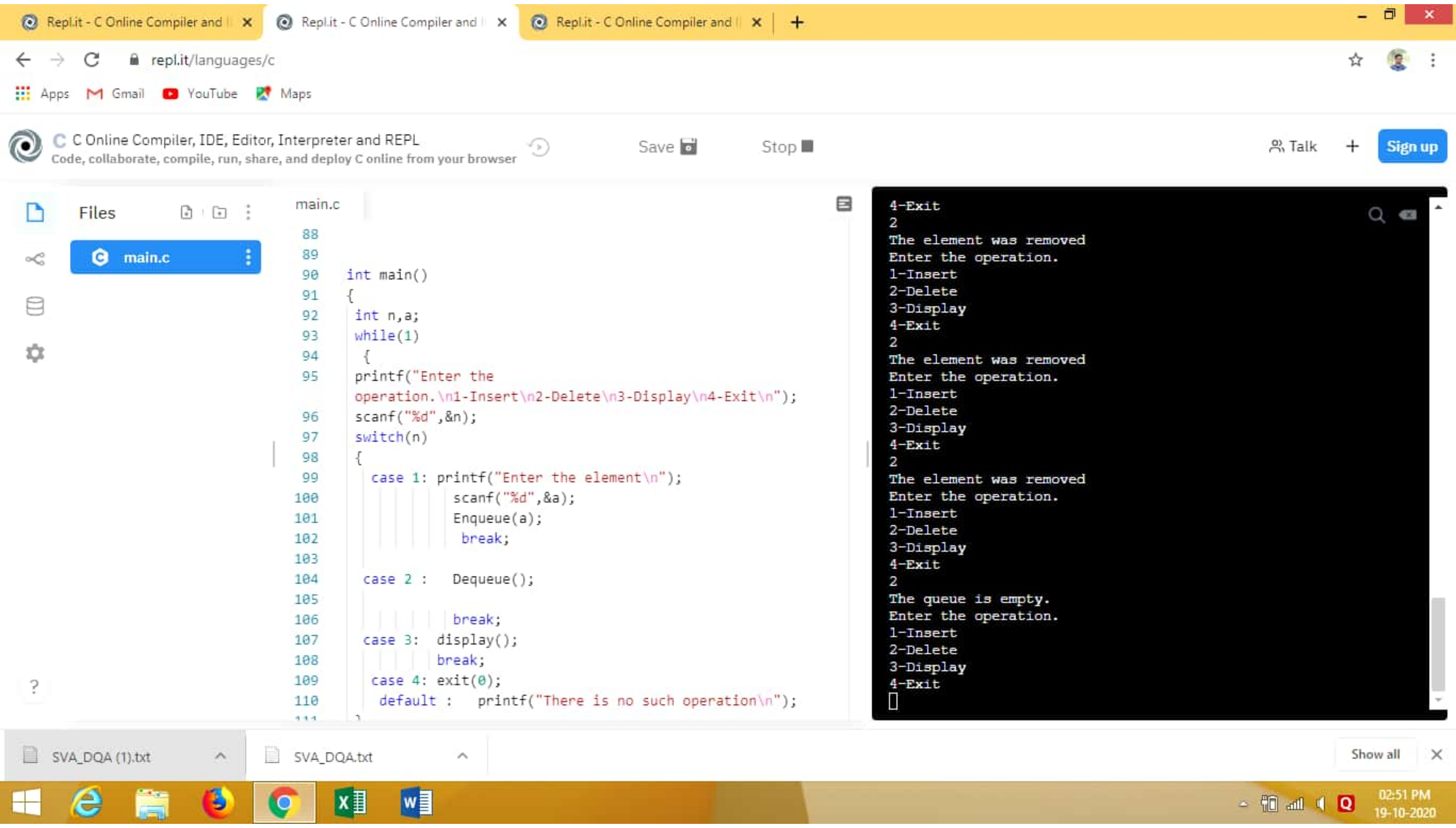
```c
        printf("\nQueue is Empty");
        return;
    }
    printf("\nElements in Circular Queue are:\n");
    if (rear >= front)
    {
        for (int i = front; i <= rear; i++)
            printf("%d\n",a[i]);
    }
    else
    {
        for (int i = front; i < SIZE; i++)
            printf("%d\n", a[i]);

        for (int i = 0; i <= rear; i++)
            printf("%d\n", a[i]);
    }
}



int main()
{
 int n,a;
 while(1)
```

```
2-Delete
3-Display
4-Exit
1
Enter the element
4
The queue is full
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
3

Elements in Circular Queue are:
1
2
3
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
2
The element was removed
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
```


Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:51 PM
19-10-2020

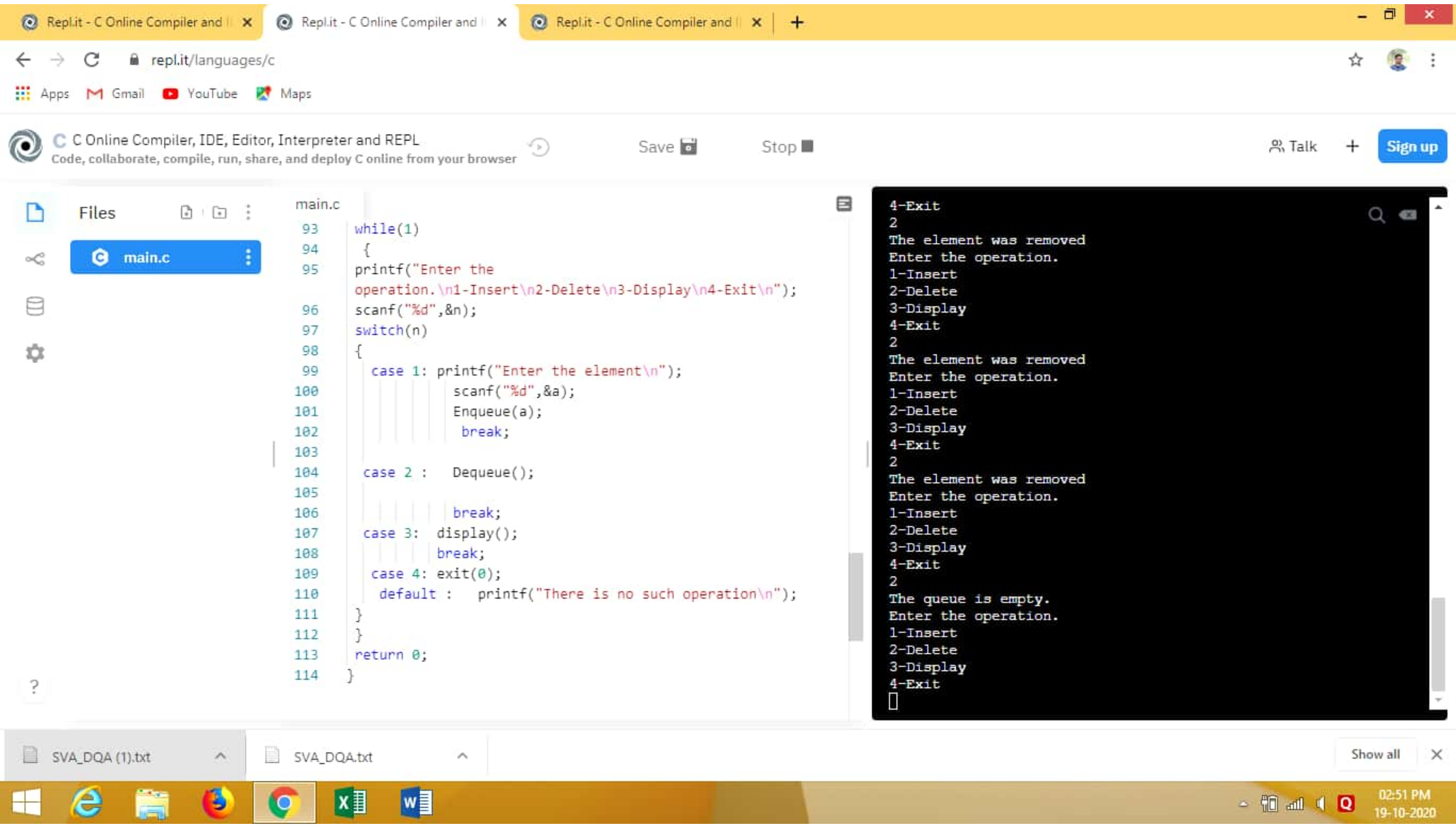
```c
            rear=-1;
             printf("The element was removed\n");
        }
        else
        {
            x=a[front];
            front=(front+1)%SIZE;
             printf("The element was removed\n");
        }
        return x;
    }


    void display()

    {
        if (front == -1)
        {
            printf("\nQueue is Empty");
            return;
        }
        printf("\nElements in Circular Queue are:\n");
        if (rear >= front)
        {
            for (int i = front; i <= rear; i++)
```

```
4-Exit
1
Enter the element
3
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
1
Enter the element
4
The queue is full
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
3

Elements in Circular Queue are:
1
2
3
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
2
```


Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:51 PM
19-10-2020

```c
int main()
{
 int n,a;
 while(1)
  {
 printf("Enter the
operation.\n1-Insert\n2-Delete\n3-Display\n4-Exit\n");
 scanf("%d",&n);
 switch(n)
 {
   case 1: printf("Enter the element\n");
            scanf("%d",&a);
            Enqueue(a);
             break;

  case 2 :   Dequeue();

            break;
  case 3:  display();
           break;
   case 4: exit(0);
    default :   printf("There is no such operation\n");
```

```
4-Exit
2
The element was removed
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
2
The element was removed
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
2
The element was removed
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
2
The queue is empty.
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
```


Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
Repl.it - C Online Compiler and I
repl.it/languages/c
Apps
Gmail
YouTube
Maps
C Online Compiler, IDE, Editor, Interpreter and REPL
Code, collaborate, compile, run, share, and deploy C online from your browser
Save
Stop
Talk
Sign up
Files
main.c
main.c
SVA_DQA (1).txt
SVA_DQA.txt
Show all
02:51 PM
19-10-2020

```c
while(1)
 {
printf("Enter the operation.\n1-Insert\n2-Delete\n3-Display\n4-Exit\n");
scanf("%d",&n);
switch(n)
{
  case 1: printf("Enter the element\n");
            scanf("%d",&a);
            Enqueue(a);
             break;

 case 2 :   Dequeue();

            break;
 case 3:  display();
          break;
  case 4: exit(0);
   default :   printf("There is no such operation\n");
}
}
return 0;
}
```

```
4-Exit
2
The element was removed
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
2
The element was removed
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
2
The element was removed
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
2
The queue is empty.
Enter the operation.
1-Insert
2-Delete
3-Display
4-Exit
```