

Lab 5: WAP to simulate the working of a queue of integers using an array. Provide the following operations: (a) Insert (b) Delete (c) Display.

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

```
#include <conio.h>
```

```
#include <process.h>
```

```
#define QUE_SIZE 3
```

```
int item, front = 0, rear = -1, q[10];
```

```
void insertrear()
```

```
{  
    if (rear == QUE_SIZE - 1)  
    {  
        printf("queue overflow\n");  
        return;  
    }  
    rear = rear + 1;  
    q[rear] = item;  
}
```

```
int deletefront()
```

```
{  
    if (front > rear)  
    {  
        front = 0;  
        rear = -1;  
        return -1;  
    }  
    return q[front++];  
}
```

```
return
```

```
return q[front++];
```

```
}
```

```
void display()
```

```
{  
    int i;  
    if (front > rear)  
    {  
        printf("queue is empty\n");  
    }  
    for (i = front; i <= rear; i++)  
        printf("%d ", q[i]);  
    printf("\n");  
}
```

return;

}

printf("contents of queue\n");

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

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

}

}

void main()

{

int choice;

for (;;) {

printf("\n 1: Insert rear\n 2: delete front\n 3: display\n 4: exit\n");

printf("enter the choice\n");

scanf("%d", &choice);

switch (choice)

{
case 1: printf("enter the item to be inserted\n");

scanf("%d", &item);

insertrear();

break;

case 2: item = deletefront();

if (item == -1)

printf("queue is empty\n");

else

printf("item deleted = %d\n", item);

break;

case 3: display();

break;

default: exit(0);

}