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

#define MAX 100
#define JOB_NAME_LEN 100

//job queue structure
char jobQueue[MAX][JOB_NAME_LEN];
int front = -1;
int rare = -1;
//function to check if queue is full
int isFull()
{
   return rare == MAX-1;
}
//function to check if  queue is empty
int isEmpty()
{
   return front == -1 || front>rare;
}
//function to add a job to queue
void addjob(char* jobName)
{
   if (isFull())
   {
      printf("queue is full. cannot add more jobs.\n");
      return;
   }
   if (isEmpty())
   {
        front = 0;
    }
    rare++;
    strcpy(jobQueue[rare], jobName);
    printf("Job '%s' added to the queue. \n", jobName);
}
//function to delete (process) a job from the queue
void deletejob()
{
    if (isEmpty())
    {
        printf("No jobs in the queue to process.\n");
        return;
    }
    printf("Job '%s' has been processed and remove from the queue.\n",jobQueue[front]);
    front++;
    if (front>rare)
    {
        //reset queue when empty
        front = rare = -1;
    }
}

//functio to display current jobs in the queue
void displayjobs()
{
    if (isEmpty())
    {
        printf("job queue is empty. \n");
        return;
    }
    printf("Current jobs in the queue:\n");
    for (int i=front; i<=rare; i++)
    {
        printf("%d. %s\n",i-front+1, jobQueue[i]);
    }
}

//Main function with menu
int main()
{
    int choice;
    char jobName[JOB_NAME_LEN];
    while (1)
    {
        printf("\n -----Job Queue Menu------- \n");
        printf("1. Add job \n");
        printf("2. Delete job\n");
        printf("3. Display jobs\n");
        printf("4. Exit \n");
        printf("Enter your choice (1-4):");
        scanf("%d",&choice);
        getchar(); //clear new libne character from input buffer
        switch (choice)
        {
            case 1:
                printf("Enter job name to add:");
                fgets(jobName, sizeof(jobName), stdin);
                jobName[strcspn(jobName,"\n")]='\0'; //remove new line
                addjob(jobName);
                break;
            case 2:
```

```
            deletejob();
            break;
          case 3:
            displayjobs();
            break;
          case 4:
            printf("Exiting Program.\n");
            exit(0);
         default:
            printf("Invalid choice. Plaease enter 1-4.\n");
      }
   }
   return 0;
}
```

```
---------------------------------------------------------------------------------------------------------------------------------
output:


-----Job Queue Menu-------
1. Add job
2. Delete job
3. Display jobs
4. Exit
Enter your choice (1-4):1
Enter job name to add:stock
Job 'stock' added to the queue.

 -----Job Queue Menu-------
1. Add job
2. Delete job
3. Display jobs
```

```
4. Exit
Enter your choice (1-4):1
Enter job name to add:order-material
Job 'order-material' added to the queue.

 -----Job Queue Menu-------
1. Add job
2. Delete job
3. Display jobs
4. Exit
Enter your choice (1-4):1
Enter job name to add:collect-material
Job 'collect-material' added to the queue.

 -----Job Queue Menu-------
1. Add job
2. Delete job
3. Display jobs
4. Exit
Enter your choice (1-4):3
Current jobs in the queue:
1. stock
2. order-material
3. collect-material

 -----Job Queue Menu-------
1. Add job
2. Delete job
3. Display jobs
4. Exit
Enter your choice (1-4):2
```

```
Job 'stock' has been processed and remove from the queue.


 -----Job Queue Menu-------

1. Add job

2. Delete job

3. Display jobs

4. Exit

Enter your choice (1-4):3

Current jobs in the queue:

1. order-material

2. collect-material


 -----Job Queue Menu-------

1. Add job

2. Delete job

3. Display jobs

4. Exit

Enter your choice (1-4):4

Exiting Program.
```