PROJECT BASED LEARNING REPORT

ON

PRINTER QUEUE SYSTEM

Submitted by:

Team Members:

1. KRISHNA AHUJA

2. JAI SURYA XESS

Under the Guidance of:

Faculty Name: Dr. RASHI SAHAY

Department of Computer Science and Engineering

Manav Rachna International Institute of Research and Studies

Academic Year: 2025–2026

ABSTRACT

The Printer Queue System is designed to simulate how print requests are managed in real-world printers. The system uses a queue data structure to store and process print jobs in **First In First Out (FIFO)** order. This project was developed using the C programming language with a linked list-based implementation of queues. It provides real-time functions such as adding print jobs, printing the next job, and viewing the list of pending jobs. This project strengthens understanding of data structures and memory management in C.

INTRODUCTION

A printer can only process one document at a time. When multiple requests are sent to the printer, they must be managed in an ordered list known as the **print queue**. This ensures fair and sequential processing of jobs. This project simulates a Printer Queue System using C programming and helps students understand the concepts of

queue operations, linked lists, dynamic memory, and user interaction in console-based applications.

OBJECTIVES

- To simulate real-world printer job management
- To implement a queue using linked lists
- To practice dynamic memory allocation using pointers
- To display and manage print jobs in FIFO order
- To strengthen understanding of data structures in C

SYSTEM ANALYSIS

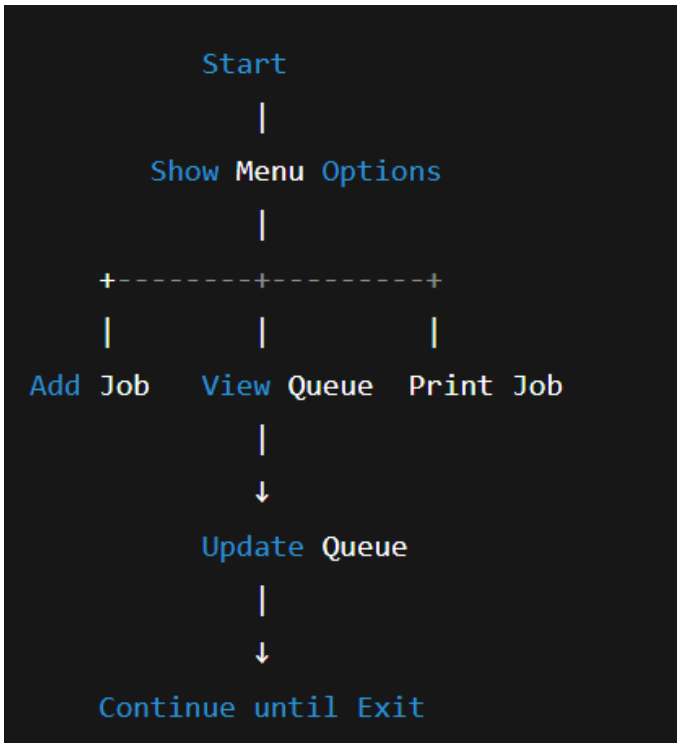
The system allows the user to:

1. **Add a print job**
2. **Print the next job**
3. **View the list of pending jobs**
4. **Handle invalid input gracefully**

This is achieved through the FIFO structure of queues where the first job added is the first job processed.

SYSTEM DESIGN

Flowchart



DATA STRUCTURE USED

The queue is implemented using a **linked list**, where each node represents a print job.

Structure Definition:

```
struct PrintJob {
    char jobName[100];
    struct PrintJob* next;
};
```

IMPLEMENTATION

Functions:

- addJob() – Adds a new print job to the queue
- printNext() – Prints and removes the job at the front of the queue
- viewQueue() – Displays all pending print jobs
- main() – Handles the menu and user input

Language / Tools Used:

- Programming Language: C
- IDE: VS Code
- Concepts: Linked List, Queue, Dynamic Memory Allocation

RESULTS AND DISCUSSION

The system allows:

- Adding multiple print jobs to the queue
- Sequential printing of jobs in FIFO order
- Viewing pending jobs at any time
- Input validation to prevent invalid operations

```
Menu:
1. Add Print Job
2. Print Next Job
3. View Print Queue
4. Exit
Enter your choice: 1

Enter job name to print: doc1
Job 'doc1' added to the queue.

Menu:
1. Add Print Job
2. Print Next Job
3. View Print Queue
4. Exit
Enter your choice: 2

Printing job: doc1

Menu:
1. Add Print Job
2. Print Next Job
3. View Print Queue
4. Exit
Enter your choice: 3

No pending print jobs.

Menu:
1. Add Print Job
2. Print Next Job
3. View Print Queue
4. Exit
Enter your choice: 4

Exiting Printer Queue System. Goodbye!
```

CONCLUSION AND FUTURE WORK

This project successfully simulates a printer queue system using the queue data structure in C. It demonstrates basic but practical usage of linked lists, memory allocation, and control flow.

Future Enhancements:

- Add job cancellation
- Introduce priority print jobs
- Develop a graphical interface
- Integrate with real printer drivers

REFERENCES

1. Let Us C – Yashavant Kanetkar
2. GeeksforGeeks – Data Structures
3. W3Schools C Programming Tutorial
4. TutorialsPoint – C Programming