

# OPERATING SYSTEM

CSE316

## FINAL PROJECT REPORT



**Topic:** File System Simulation

**Submitted By:**

**To:**

Ishaant Kumar Singh

Section - K22UP

Roll no. : 47

**Submitted**

Cherry Khosla

UID: 13436

# Declaration

I hereby declare that the project work entitled (“File System Simulation”) is an authentic record of our own work carried out as requirements of Project for the award of B.Tech degree in Computer Science and Engineering (AI & ML) from Lovely Professional University, Phagwara, under the guidance of **Cherry Khosla**. All the information furnished in this project report is based on our own intensive work and is genuine.

Name of Student : Ishaant Kumar Singh

Registration Number: 12203987

Question No. : 9

---

Signature of Student  
Date: 31/10/23

# Certificate

This is to certify that the declaration statement made by this student is correct to the best of my knowledge and belief. He has completed this Project under my guidance and supervision. The present work is the result of their original investigation, effort and study. No part of the work has ever been submitted for any other degree at any University. The Project is fit for the submission and partial fulfillment of the conditions for the award of B.Tech degree in Computer Science and Engineering (AI & ML) from Lovely Professional University, Phagwara.

**Signature and Name of the Mentor:**

**Designation:**

**School of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab.**

**Date :**

# Acknowledgement

I would like to extend my sincere thanks to the Lovely School of Computer Science and Engineering for providing me with the opportunity to fulfill my wish and achieve my goal. I am grateful to **Cherry Khosla** mam for providing me with the opportunity to undertake this project and for providing me with all the necessary facilities. I am highly thankful to sir for his active support, valuable time and advice, whole-hearted guidance, sincere cooperation, and pain-taking involvement during the study and in completing the assignment of preparing the said project within the time stipulated. Lastly, I am thankful to all those, particularly the various friends, who have been instrumental in creating a proper, healthy, and conducive environment and including new and fresh innovative ideas for me during the project. Without their help, it would have been extremely difficult for me to prepare the project in a timebound framework.

## Question No. :- 9

Scenario: A company has a large number of employees who work on various projects and create different types of files such as documents, spreadsheets, and images. The company wants to implement a new file system that can manage the disk space efficiently and handle the high volume of file operations.

Problem: Create a simulation program that simulates a file system for the company. The program should include a file system manager that uses a specific file allocation algorithm (e.g. Contiguous Allocation or Linked Allocation) to allocate space for files on a simulated disk. The students should also include a mechanism for deleting, renaming and moving files. The simulation should run for a set amount of time and record the average amount of fragmentation and the number of wasted disk blocks at the end of each time unit.

The students should also consider the following factors in their simulation:

- The employees will be creating and editing different types of files (e.g. documents, spreadsheets, images) with varying file sizes.
- The employees will be frequently adding and deleting files.
- The company wants to minimize the amount of wasted disk space.

At the end of the simulation, the students should provide a report on their findings and observations, including the performance of the file system under different scenarios and the trade-offs involved in the different file allocation algorithms.

Expected outcomes:

1. File allocation: The program should demonstrate the ability to allocate space for files on a simulated disk using the chosen file allocation algorithm (e.g. Contiguous Allocation or Linked Allocation).
2. File deletion and renaming: The program should include a mechanism for deleting and renaming files.
3. Disk fragmentation: The program should record the average amount of fragmentation at the end of each time unit. Fragmentation occurs

when there are small, unused blocks of disk space scattered throughout the disk space.

4. Wasted disk space: The program should also record the number of wasted disk blocks at the end of each time unit. Wasted disk space refers to blocks of disk space that are no longer being used by any files.

5. Simulation results: The program should run the simulation for a set amount of time and display the results, including the average amount of fragmentation and the number of wasted disk blocks, at the end of each time unit. The students should also experiment with different input scenarios and algorithms to compare the results and see how different factors affect the performance of the file system.

## **Introduction**

In the contemporary corporate landscape, efficient data management stands as a pivotal concern for companies grappling with vast volumes of digital information. This project embarks on the development of a simulation program tailored to simulate a robust file system capable of handling a multitude of file types, optimizing disk space, and streamlining file operations. This endeavor is undertaken in response to the challenge posed by a hypothetical company that must effectively manage a large workforce generating a myriad of documents, spreadsheets, images, and other file formats.

This project aims to explore file allocation algorithms as the central tenet of modern file systems. We will implement specific allocation methods, such as Contiguous Allocation or Linked Allocation, in our simulated file system, shedding light on their performance and trade-offs.

The simulation program will encompass the complete file management lifecycle, including file creation, deletion, and renaming. This ensures that the file system manager we develop efficiently allocates space and adapts to the organization's evolving needs.

The focal point of this project is to address fragmentation and wasted disk space issues. We will record and analyze the average amount of fragmentation and the number of wasted disk blocks at the end of each time unit, offering quantitative insights into the file system's effectiveness.

Through this simulation, we will provide valuable insights into the performance and adaptability of the file system under different conditions. This research will serve as a reference for organizations seeking to implement or enhance their file systems and make informed decisions on file allocation strategies and management practices. The results will empower companies with the knowledge to optimize data management, reduce inefficiencies, and enhance productivity in the data-driven business landscape.

## **Requirement of the Solution**

**1. File Allocation Algorithms:** The solution implements one or more file allocation algorithms, such as Contiguous Allocation or Linked Allocation, to efficiently allocate disk space for files based on specific needs and constraints.

**2. File Creation and Storage:** Users can create files by specifying a name and size (in kilobytes) and store them on the simulated disk, handling files of varying sizes.

**3. File Deletion:** Users can delete files from the disk by specifying the file's name, freeing up the space for future use.

**4. File Renaming:** The solution provides the capability to rename files on the disk. Users can specify the old and new names of the file to be renamed.

**5. Disk Initialization:** The system offers a mechanism to initialize the disk by clearing all existing files and returning all disk blocks to an empty state.

**6.Disk Status Display:** Users can view the current status of the disk, showing which blocks are occupied by files and providing information about file names and sizes.

**7.Fragmentation Tracking:** The program monitors and calculates the amount of fragmentation on the disk, displaying this fragmentation information.

**8. Wasted Disk Space Measurement:** The system measures and reports the number of wasted disk blocks, referring to blocks that are no longer in use by any files but remain unallocated.

## Step Wise Pseudo Code

### 1. Initialize the Filesystem

- Create a Filesystem structure, including files, directories, and disk blocks.
- Initialize all disk blocks as unused.

2. Set the current directory to the root directory.

### 3. Display a menu and start a loop for user interaction.

- Display menu options and request user input.

### 4. Process user input based on the selected menu option:

- When the user selects "Store a file":
  - Prompt the user for a file name and size.
  - Call `findFreeBlocks` to find contiguous free blocks for the file.
  - Update the filesystem and store the file.



- b. When the user selects "List contents of the current directory":
  - i. Display the contents of the current directory, including files and subdirectories.
- c. When the user selects "Create a subdirectory":
  - i. Prompt the user for a subdirectory name.
  - ii. Create a new subdirectory and add it to the current directory.
- d. When the user selects "Navigate to a different directory":
  - i. Prompt the user for the directory name or ".." to move up.
  - ii. Update the current directory based on user input.
  - iii. Maintain a directory history using a stack.
- e. When the user selects "Rename a file":
  - i. Prompt the user for the old and new file names.
  - ii. Call ``renameFile`` to rename the file.
- f. When the user selects "Rename a directory":
  - i. Prompt the user for the old and new directory names.
  - ii. Call ``renameDirectory`` to rename the directory.
- g. When the user selects "Defragment the disk":
  - i. Call ``calculateDiskFragmentation`` to calculate fragmentation and wasted space.
  - ii. Display the results.
- h. When the user selects "Calculate Wasted Disk Space":
  - i. Call ``calculateWastedDiskSpace`` to calculate wasted disk space.
  - ii. Display the result.
- i. When the user selects "Delete a file":
  - i. Prompt the user for the file name to delete.
  - ii. Call ``deleteFile`` to remove the file and free disk blocks.
- j. When the user selects "Delete a directory":
  - i. Prompt the user for the directory name to delete.

- ii. Call ``deleteDirectory`` to delete the directory if it's empty.
  - k. When the user selects "Exit":
    - i. End the program and exit the loop.
  - l. For any other input, display an error message and allow the user to try again.
- 5.** Continue looping until the user chooses to exit.
- 6.** Upon program exit, provide a summary of the file system state, and save any necessary data to persist the changes.
- 7.** End the program.

## **GitHub Link Of The Project**

**Link** - <https://github.com/ishaant-97/fileManagementSystem-OS-Project>

# SnapShot of the Output

```
File System Simulation Menu:
```

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

```
Enter your choice: 1
```

```
Enter file name: file1
```

```
Enter file size (in KB): 5
```

```
File stored successfully: file1
```

```
-----
```

```
-----  
File System Simulation Menu:
```

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

```
Enter your choice: 2
```

```
Contents of Directory: Root
```

```
    File: file1 (Size: 5 KB)
```

File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 3

Enter subdirectory name: dir1

File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 4

Enter directory name to navigate to (or use '..' to go up): dir1

File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 5

Enter the name of the file to rename: file1

Enter the new name for the file: file8

File renamed successfully to: file8

File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 6

Enter the name of the directory to rename: dir1

Enter the new name for the directory: dir8

Directory renamed successfully to: dir8

File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 7

Disk Fragmentation: 0 out of 5 blocks.

File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 8

Wasted Disk Space: 995KB



File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 9

Enter the name of the file to delete: file8

File 'file8' deleted successfully.

File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 10

Enter the name of the directory to delete: dir8

Directory 'dir8' deleted successfully.

File System Simulation Menu:

1. Store a file
2. List contents of the current directory
3. Create a subdirectory
4. Navigate to different directory
5. Rename a file
6. Rename a directory
7. Defragment the disk
8. Calculate Wasted Disk Space
9. Delete a file
10. Delete a directory
11. Exit

Enter your choice: 11

Exiting the program.