Visvesvaraya Technological University

Belagavi, Karnataka

ವಿಶ್ವೇಶ್ವರಯ್ಯ ತಾಂತ್ರಿಕ ವಿಶ್ವವಿದ್ಯಾಲಯ

ಬೆಳಗಾವಿ, ಕರ್ನಾಟಕ

A FILE STRUCTURE MINI- PROJECT [18ISL67] REPORT

ON

"PRISON MANAGEMENT SYSTEM USING RRN"

Submitted to Visvesvaraya Technological University in partial fulfillment of the requirement for the award of Bachelor of Engineering degree in Information Science and Engineering.

Submitted by

Name	**USN**
1. Abhishek J M	**4JN18IS001**
2. Adarsh N Inamdar	**4JN18IS002**

Under the guidance of

Mr. Sharath Kumar S R B.E,M.Tech,
Asst Prof, Dept. of IS&E,
JNNCE, Shivamogga

Department of Information Science & Engineering
Jawaharlal Nehru National College of Engineering
Shivamogga - 577 204

Aug 2021

National Educational Society ®

Jawaharlal Nehru National College of Engineering

Department of Information Science & Engineering

CERTIFICATE

This is to certify that the FILE STRUCTURE [18ISL67] - MINI project entitled

“PRISON MANAGEMENT SYSTEM USING RRN”

Submitted by

Name	USN
1. Abhishek J M	4JN18IS001
2. Adarsh N Inamdar	4JN18IS002

Students of 6th semester B.E. ISE, in partial fulfillment of the requirement for the award of degree of Bachelor of Engineering in Information Science and Engineering of Visvesvaraya Technological University, Belagavi during the year 2018-19

Signature of Guides	Signature of HOD
______________________________	________________________
Mr. Sharath Kumar S RB.E.,M.Tech	**Dr. Sanjeev Kunte R** **M.Tech, Ph.D.**
Asst. Prof, Dept. of IS&E,	Prof. & Head, Dept. of IS&E

Examiners: **1.** ________________________ **2.** ________________________

ABSTRACT

This project is aimed at developing a prison management system that is the collection of details of the prisoner and his details.

Basically prison is a correctional institution where persons are confined while on trial or for punishment. Each prison may have hundreds of criminals. With hardcopy of records it is hard to search report or record about a particular criminal in prison. The purpose of this document is to present a detailed description of the Prison Management System. The document will describe what to do, the constraints under which it must operate and how the system will react to external stimuli. This software will be a Prison Management System for Prison Department. This system will be efficient collection of records and details about the prisoners in prison.

ACKNOWLEDGEMENT

On presenting the File Structure [18ISL67] Mini – Project report on “Bus Timings management using RRN”, I feel great to express my humble feelings of thanks to all those who have helped me directly or indirectly in the successful completion of the project work.

I would like to thank our respected guide Mr. Sharath Kumar S R **, Asst. Prof, Dept. of IS & E who has helped me a lot in completing this task, for their continuous encouragement and guidance throughout the project work.**

I would like to thank Dr. Sanjeev Kunte R**, Professor and Head, Dept of ISE, JNNCE, Shivamogga and** Dr. Manjunatha P, **The Principal, JNNCE, Shivamogga for all their support and encouragement.**

I am grateful to Department of Information Science and Engineering and our institution Jawaharlal Nehru National College of Engineering and for imparting me the knowledge with which I can do my best.

Finally, I also would like to thank the whole teaching and non-teaching staff of Information Science and Engineering Dept.

Thanking you all,

Abhishek J M	**4JN18IS001**
Adarsh N Inamdar	**4JN18IS002**

TABLE OF CONTENTS

CHAPTER-1
INTRODUCTION

1.1 RECORD

A record is a collection of fields, possibly of differ data types, typically in fixed number and sequence. The fields of a record may also be called members, particularly in oriented programming. Fields may also be called elements, through the risk confusion with the elements of the collection.

For example, a data could be stored as record containing a numeric year field, a month field represented as a string, and a numeric day-of-month field. A personnel record might contain a name, a salary, and a rank. A circle of record contains center and a radius in this instance, the center itself might be represented as point record containing x and y coordinates.

Record are distinguished from arrays be the fact that their number of fields is typically fixed, each field has a name, and that field may have a different type.

A record type is a data type that describes such values and variable. Most modern computer languages allow the programmer to define new data record types. The definition includes specifying the data types of each field and the identified by which it can be accessed. In type theory, product types are generally preferred due to the simplicity, but the proper record types are studied in languages such as System F-sub. Since type-theoretical records may contain first-class function-typed fields in addition to data, they can be express many features of object-oriented programming.

Records can exist in any storage medium, including main memory and mass storage devices such as magnetic tapes or hard disk. Records are the fundamental component of most Structure, especially liked data structure. Many computer files are organized as array of logical records, often grouped into larger physical records or block for efficiency.

1.1.1 RECORD DEFINITION

A record can be defined as set of fields that belongs together when the file is viewed in term of higher level of organization. For example we can define structure as follows:

```
Struct date
{
int year;
int month;
int day;
};
```

1.2 TYPES OF RECORDS

1.2.1 Fixed length records

1.2.2Variable length records

1.2.1 Fixed Length Record:

Each record is stored in fixed size. The size can be determined by adding the maximum space occupied by each field and some space reserved for the header data.

1.2.2 Variable Length Record and Keys

In many applications information associated with a key varies in length. Secondary indexes that reference inverted lists are one of excellent example for this. One way to handle this variability is to place the associated information in a separate, variable-length record file. Another approach is to allow a variable number of keys and records in a B-tree page.

1.3Relative Record Number (RRN)

It's an integer that is used to represent a fixed length record. RRN starts with followed by 1, 2, 3…… where RRN 0 represent 1st record. RRN 1 represents-2nd record and so on.
The Relative Record Number identifies which position in a file the record is in. If, for example, a record has the RRN of 10 there does not have to be nine records before it. When a record is deleted its space in the file is retained, and it is not freed until the file is reorganized using the RGZPFM command, or if the file is set to reuse deleted records a new record is added to the file. We can use RRN options on most commands that access data sets. RRN specifies that the record identification field contains the relative record number of the record to be accessed. RRN starts with 0 followed by 1, 2, 3…. Where RRN 0 represents first record, RRN 1 represents second record and so on. If length of a record is N and RRN of that record is R then address of that record is calculated as R*N.

1.4Relative record Data Set (RRDS)

A relative record data set is a type of data set organization used by the VSAM computer data storage system. Records are accessed based on their ordinal position in the file (RRN, relative record number). For example, the desired record to be accessed might be the 42ndrecord in the file out of 999total.

The concept of RRDS is similar to sequential access method, but it can access with data in random access and dynamic access.

1.5 RRDS Structure

An RRDS consist of data records in sequence, with the record number indicating the records the logical position in the data set. A program can access records randomly using this positional number or access record sequentially. But unlike a Key Sequenced Data Set, an RRDS has no keys, so the program cannot access records by key values. Keys may be used to access records in an RRDS by defining an alternate index.

1.6 Key Sequenced Data Set (KSDS)

A Key Sequenced Data Set (KSDS) is a type of data set used by the IBM VSAM computer data storage system. Each record in a KSDS data file is embedded with unique key. A KSDS consists of two parts, the data component and a separate index file known as the index component which allows the system to physically locate the record in the data file by its key value. Together, the data and index components are called a cluster.

1.7 Entry Sequenced Data Set (ESDS)

An Entry Sequenced Data Set (ESDS) is a type of data set used by the VSAM computer data storage system. Records are accessed based on their sequential order that is the order in which they were written to the file. Which means that accessing a particular record involves searching all the records sequentially until it is located, or by using a relative physical address(Relative Byte Address, RBA),that is the number of bytes from the beginning of file to start reading.

1.8 Direct Access

- With direct access, we can seek directly to the beginning of the record.

  Time is in O (1) for n records.
- Index may be the beginning of the required record.
- We know the relative record number RRN

First record has RRN 0, the next has RRN 1, and so forth.

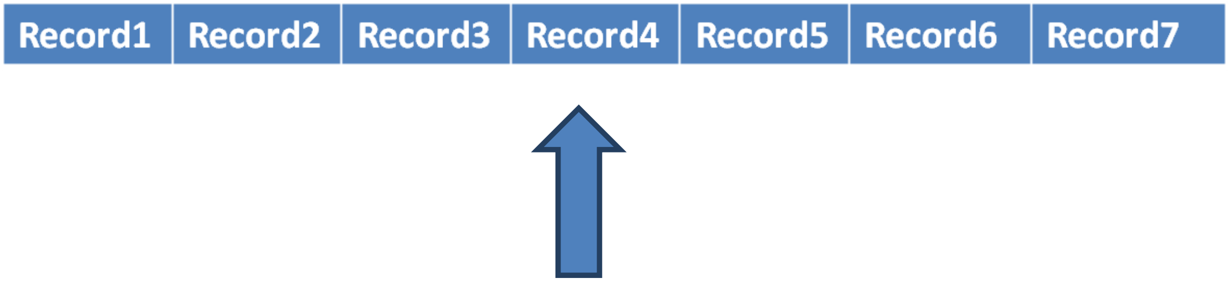


Fig 1.1 DIRECT ACCESS

1.9 Direct Access by RRN

RRN is not useful when working with variable length record

The access is still sequential.

To get the record we want, we have to read sequentially through the file, counting records as we go.

Time is in O (n).

With Fixed length record RRN is useful,

If the records are all the same length, we can use records RRN to calculate byte offset of the start of the record relative to the start of the file.

Example: RRN=30, record length=100bytes:
byte offset=30*100=3000

CHAPTER-2

2.1 DESIGN OF PRISON MANAGEMENT SYSTEM

Prison management concerns in developing a prison management system that is the collection of details of the prisoner and his details.

The records in the program are **inmate, prison.**

2.1.1 INMATE RECORDS

In the inmate record, the fields are prison_id, inmate_id, inmate_name, nationalism, crime, health_status, datein

prison_id	inmate_id	inmate_name	nationalism	crime	health_status	datein

2.1.2 PRISON RECORDS

In the prison records, the fields are prison_id, prison_name.

prison_id	prison_name

CHAPTER 3

ALGORITHM AND IMPLEMENTATION

This chapter algorithm and implementation of the Prison Management System using file structure concept of relative record number.

3.1 ALGORITHM

3.1.1 MAIN ( ) FUNCTION

It will give the choice to select among inmate, prison. Once the choice is selected it gives further choice to insert a record or search a record.

```
   1. In a while loop
      Print Prison management system using rrn
      Print enter the choice for inmate, prison details
      Switch (expression)
      Case 1:  In a do while loop
               Print prison operations.
               If the choice is 1, call read(), pack(), write_to_file() method.
               If the choice is 2, call display_rrn_list()
               If the choice is 3, call create_rrn_list(), search method.
               If the choice is 4, call remove_rrn_list(), remove method.
               If the choice is 5, go to home.
      Case 2:  In a do while loop
               Print inmate operations.
               If the choice is 1, call read1(), pack1(), write_to_file1() method.
               If the choice is 2, call display_rrn_list1()
               If the choice is 3, call create_rrn_list1(), search method.
               If the choice is 4, call remove_rrn_list1(), remove method.
               If the choice is 5, go to home.
      Case 3:Exit
2. End while
3. End
```

read()

Read the bus details by asking the input from the user like:

```
1. Read prison_id, prison_name.
2. End
```

pack( )

Store all the volumes of bus details

```
1. Erase the buffer garbage value
```

```
    2. Store prison_id, prison_name in the buffer along with the delimiter between
    them. End of the record is identified by $ symbol.

    3. End
```

write_to_file( )

```
1. Open a txt file in the output and in append mode.
2. Redirect contents of buffer to file.
3. Close the file.
4. End
```

Create_rrn_list()

```
1. Initialize rrn to 1 and no_of_records to 0.
2. Open a file filename.txt in input mode.
3. Print RRN no Record.
4. In do while loop
a. get the current position of pointer, store in pos.
b. and initialize pos to rrn_array.
c. get the contents of buffer.
d. if end of the file reached, break.
e. print rrn and buffer
f. increment rrn
5. Until true.
6. End do while loop.
7. close the file.
8. End.
```

Search() function:

```
 1. If rrn_no is negative or zero or greater than no_of_recs then print record not found.
 2. Else
     a. Put the volume of rrn_array to pos.
     b. Open the file prison.txt in input mode.
     c. move the get pointer to beginning of pos.
     d. print record found.
```

```
        e. get the content of buffer.
        f. print the buffer content.
        g. close the file.
  3. End if
  4. End
```

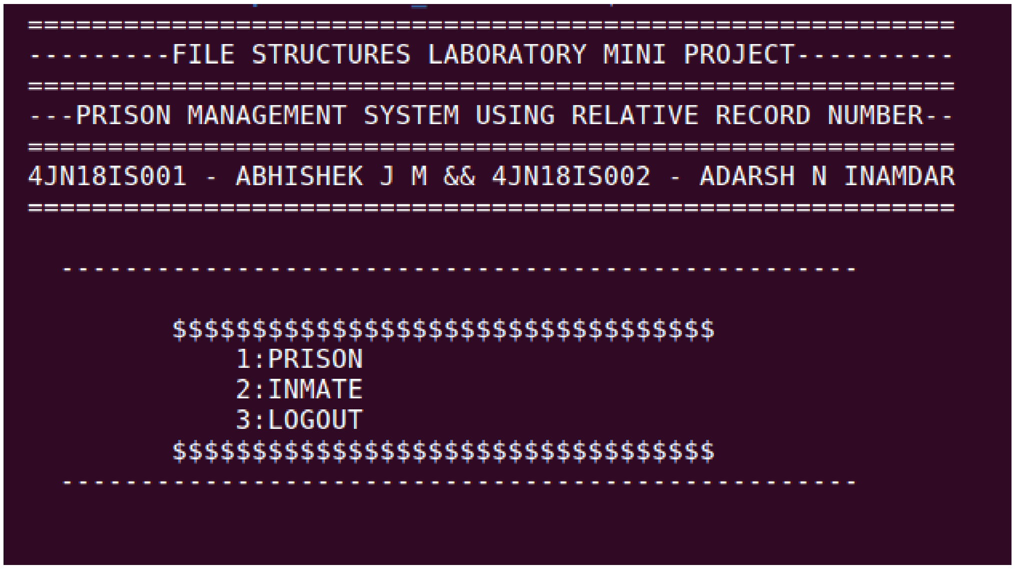
Display() function

1. Creates the RRN for each record in prison.txt file.
2. And unpacks it in the console.

Remove() function

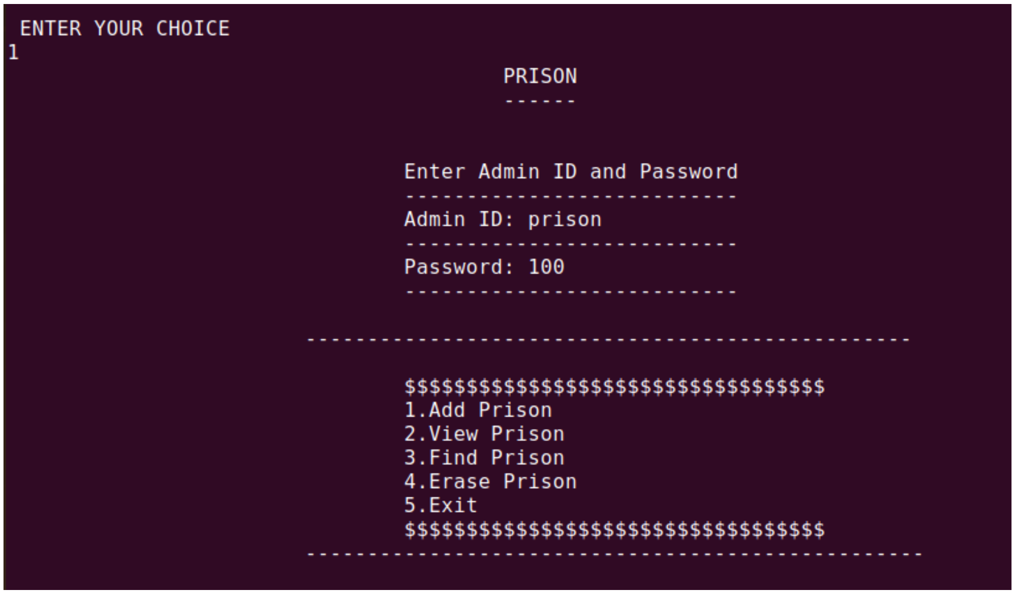
1. First rrn is created for the set of records in prison.txt file.
2. Enter the rrn no corresponding to the record which has to be deleted.
3. Call search function to search the rrn no.
4. Have the * value corresponding to the record that has to be removed.

CHAPTER 4
RESULT ANALYSIS AND OBSERVATION



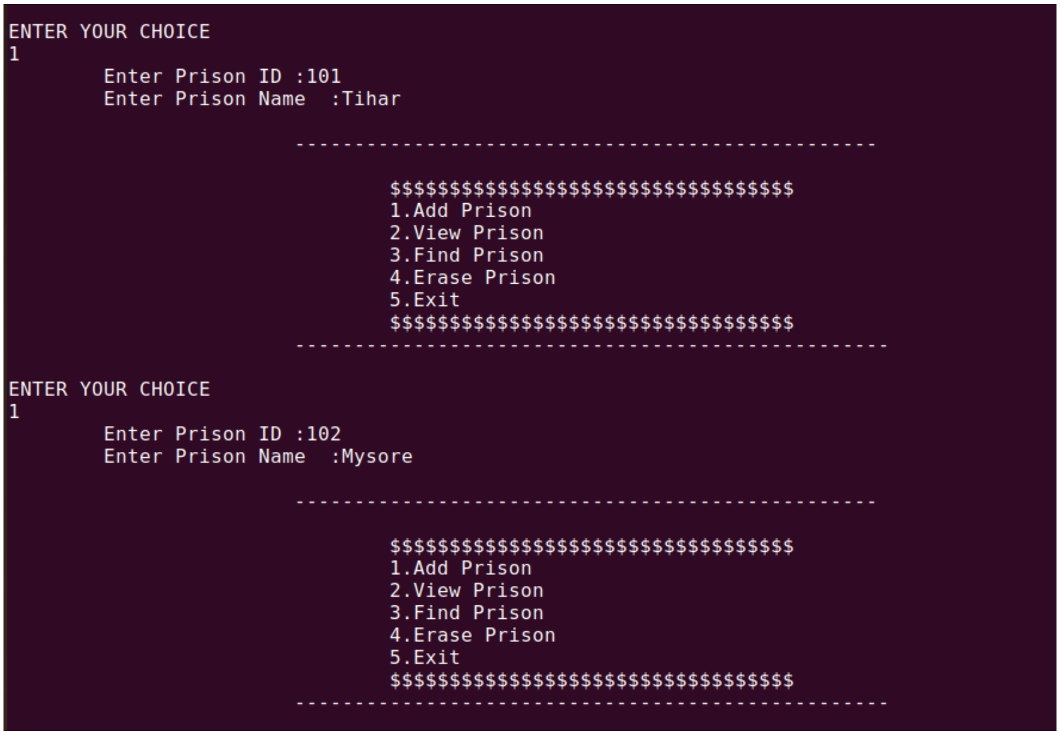
```
==========================================================
---------FILE STRUCTURES LABORATORY MINI PROJECT----------
==========================================================
---PRISON MANAGEMENT SYSTEM USING RELATIVE RECORD NUMBER--
==========================================================
4JN18IS001 - ABHISHEK J M && 4JN18IS002 - ADARSH N INAMDAR
==========================================================

  --------------------------------------------------


         $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
             1:PRISON
             2:INMATE
             3:LOGOUT
         $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
  --------------------------------------------------
```

Fig 1: Main Menu



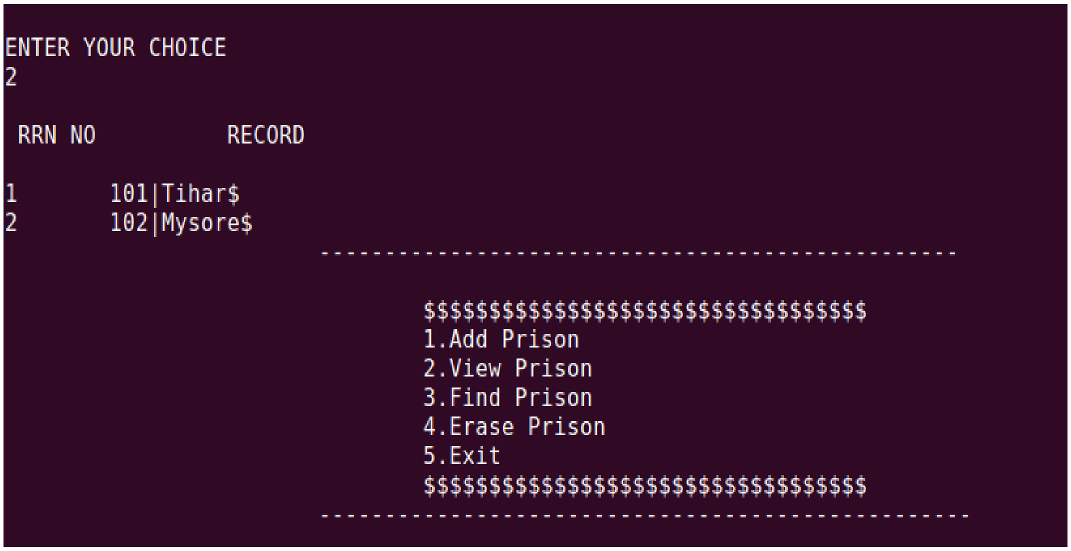
```
 ENTER YOUR CHOICE
1
                                        PRISON
                                        ------


                                Enter Admin ID and Password
                                ---------------------------
                                Admin ID: prison
                                ---------------------------
                                Password: 100
                                ---------------------------

                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Prison
                                2.View Prison
                                3.Find Prison
                                4.Erase Prison
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------
```

Fig 2: Prison User Authentication



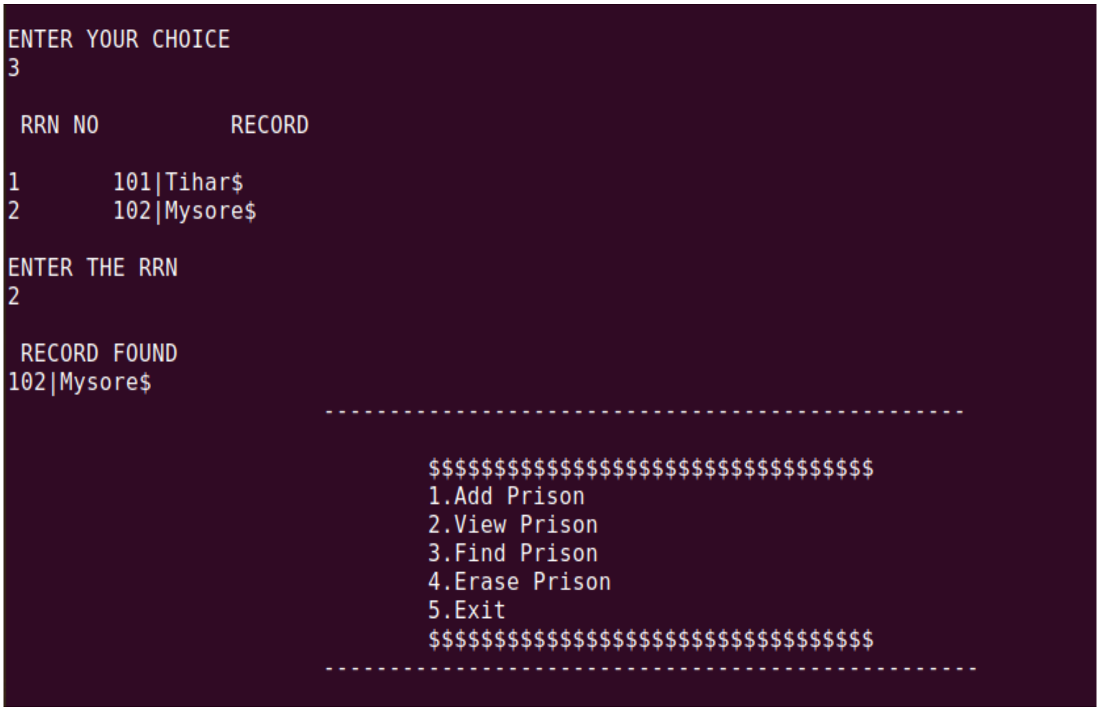
```
ENTER YOUR CHOICE
1
        Enter Prison ID :101
        Enter Prison Name  :Tihar

                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Prison
                                2.View Prison
                                3.Find Prison
                                4.Erase Prison
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------

ENTER YOUR CHOICE
1
        Enter Prison ID :102
        Enter Prison Name  :Mysore

                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Prison
                                2.View Prison
                                3.Find Prison
                                4.Erase Prison
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------
```

Fig 3: Insertion



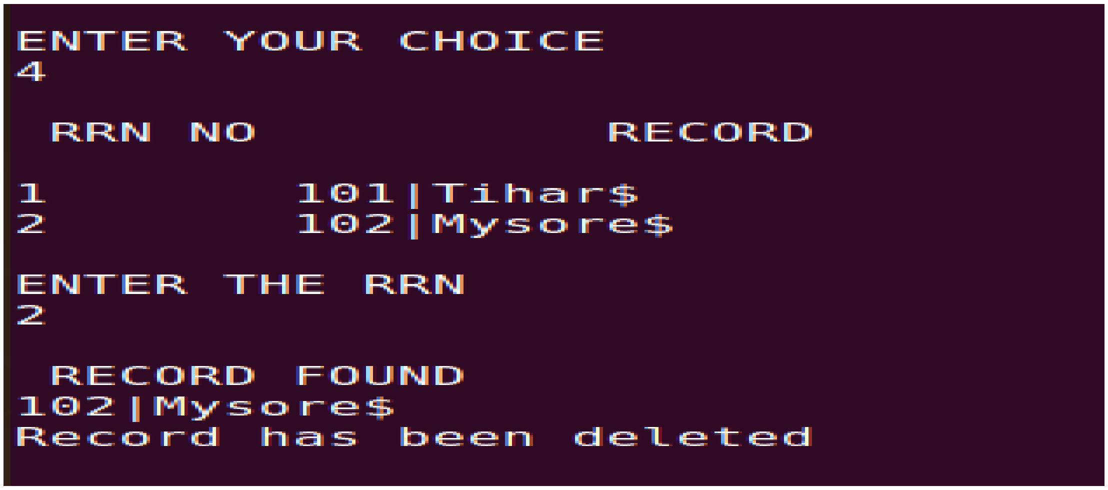
```
ENTER YOUR CHOICE
2

 RRN NO          RECORD

1       101|Tihar$
2       102|Mysore$
                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Prison
                                2.View Prison
                                3.Find Prison
                                4.Erase Prison
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------
```

Fig 4: Display



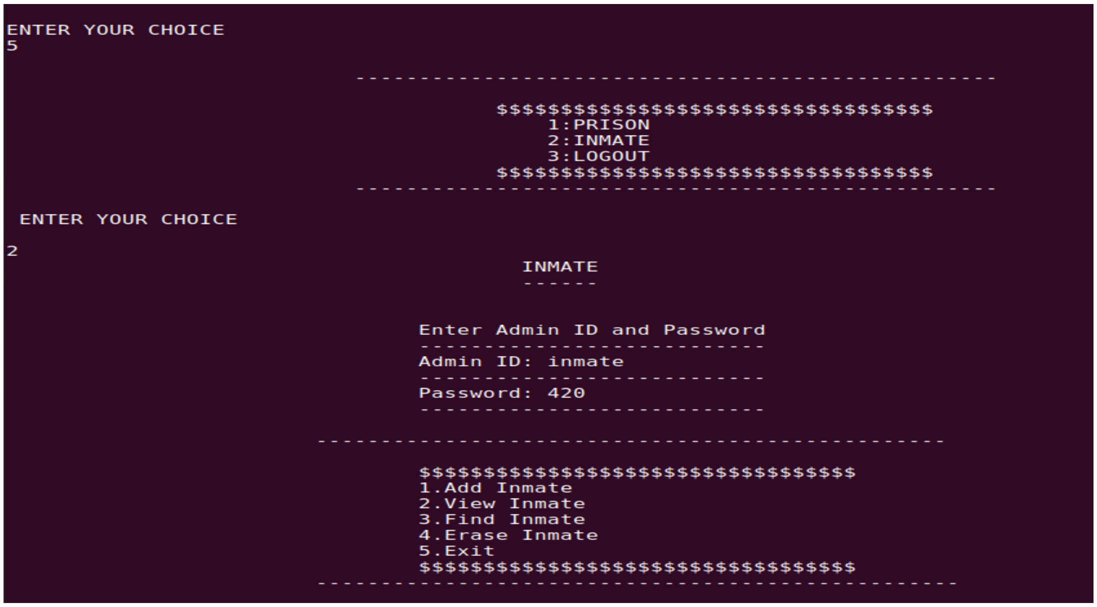
```
ENTER YOUR CHOICE
3

 RRN NO          RECORD

1       101|Tihar$
2       102|Mysore$

ENTER THE RRN
2

 RECORD FOUND
102|Mysore$
                        ------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Prison
                                2.View Prison
                                3.Find Prison
                                4.Erase Prison
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        -------------------------------------------------
```

Fig 5: Search



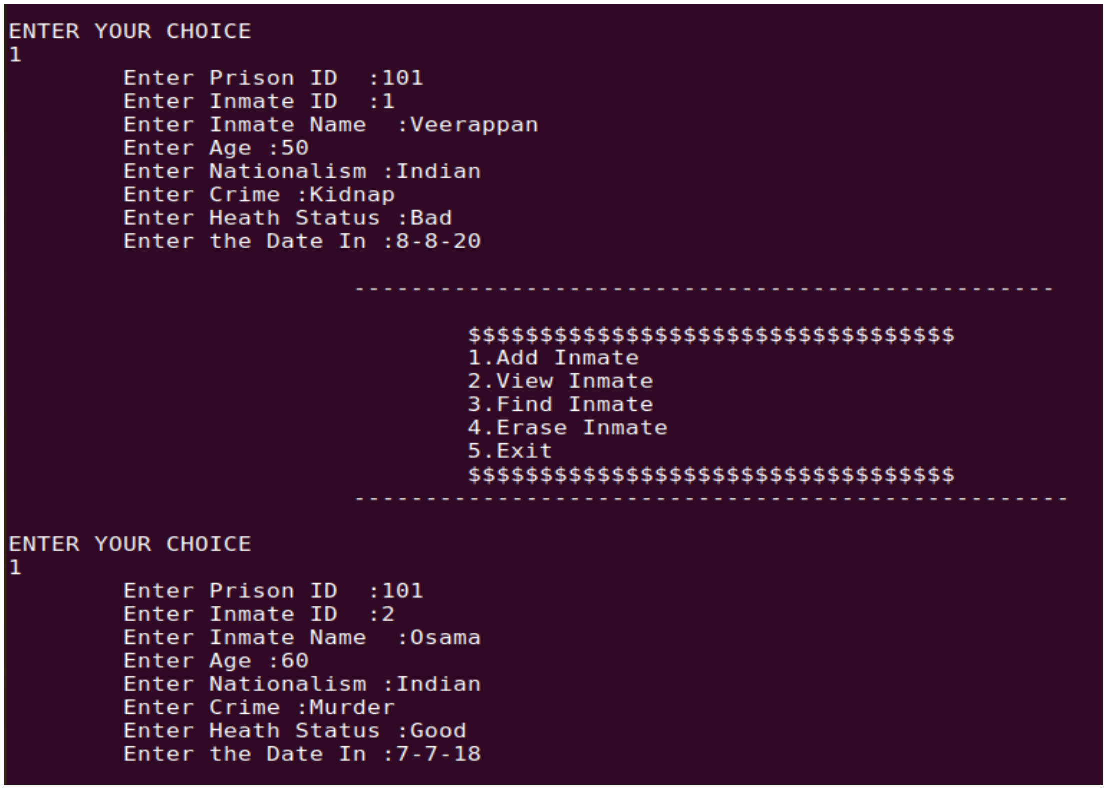
```
ENTER YOUR CHOICE
4

 RRN NO              RECORD

1          101|Tihar$
2          102|Mysore$

ENTER THE RRN
2

 RECORD FOUND
102|Mysore$
Record has been deleted
```

Fig 6: Deletion



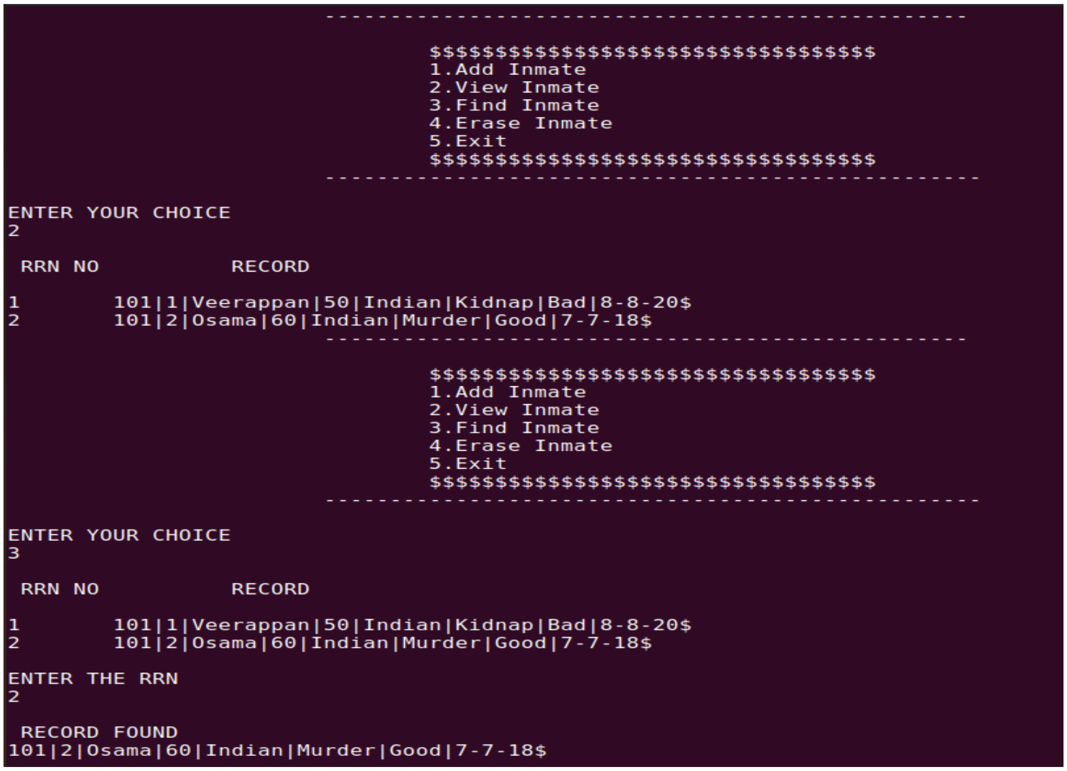
```
ENTER YOUR CHOICE
5
                           --------------------------------------------------

                                      $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                          1:PRISON
                                          2:INMATE
                                          3:LOGOUT
                                      $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                           --------------------------------------------------

 ENTER YOUR CHOICE

2
                                        INMATE
                                        ------


                                Enter Admin ID and Password
                                ---------------------------
                                Admin ID: inmate
                                ---------------------------
                                Password: 420
                                ---------------------------

                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Inmate
                                2.View Inmate
                                3.Find Inmate
                                4.Erase Inmate
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------
```

Fig 7: Inmate User Authentication



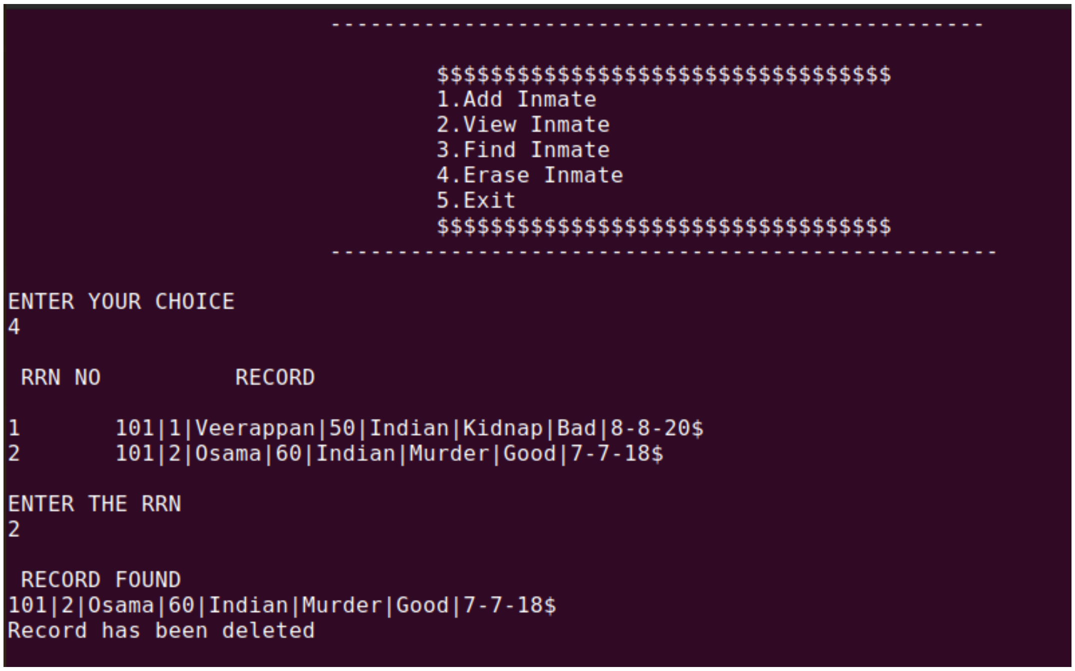
```
ENTER YOUR CHOICE
1
        Enter Prison ID  :101
        Enter Inmate ID  :1
        Enter Inmate Name  :Veerappan
        Enter Age :50
        Enter Nationalism :Indian
        Enter Crime :Kidnap
        Enter Heath Status :Bad
        Enter the Date In :8-8-20

                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Inmate
                                2.View Inmate
                                3.Find Inmate
                                4.Erase Inmate
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------

ENTER YOUR CHOICE
1
        Enter Prison ID  :101
        Enter Inmate ID  :2
        Enter Inmate Name  :Osama
        Enter Age :60
        Enter Nationalism :Indian
        Enter Crime :Murder
        Enter Heath Status :Good
        Enter the Date In :7-7-18
```

Fig 8: Insertion



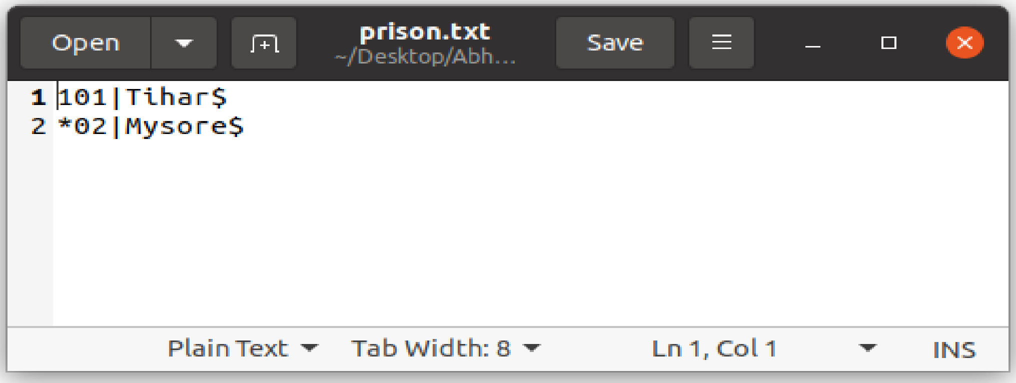
```
                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Inmate
                                2.View Inmate
                                3.Find Inmate
                                4.Erase Inmate
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------

ENTER YOUR CHOICE
2

 RRN NO          RECORD

1       101|1|Veerappan|50|Indian|Kidnap|Bad|8-8-20$
2       101|2|Osama|60|Indian|Murder|Good|7-7-18$
                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Inmate
                                2.View Inmate
                                3.Find Inmate
                                4.Erase Inmate
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------

ENTER YOUR CHOICE
3

 RRN NO          RECORD

1       101|1|Veerappan|50|Indian|Kidnap|Bad|8-8-20$
2       101|2|Osama|60|Indian|Murder|Good|7-7-18$

ENTER THE RRN
2

 RECORD FOUND
101|2|Osama|60|Indian|Murder|Good|7-7-18$
```

Fig 9: Display & Search



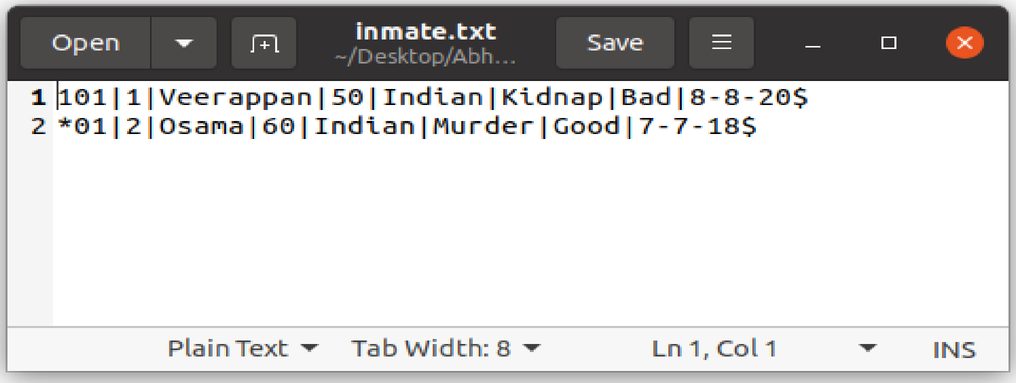
```
                        -------------------------------------------------

                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                                1.Add Inmate
                                2.View Inmate
                                3.Find Inmate
                                4.Erase Inmate
                                5.Exit
                                $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                        --------------------------------------------------

ENTER YOUR CHOICE
4

 RRN NO          RECORD

1       101|1|Veerappan|50|Indian|Kidnap|Bad|8-8-20$
2       101|2|Osama|60|Indian|Murder|Good|7-7-18$

ENTER THE RRN
2

 RECORD FOUND
101|2|Osama|60|Indian|Murder|Good|7-7-18$
Record has been deleted
```

Fig 10: Deletion



```
101|Tihar$
*02|Mysore$
```

Fig 11: Prison txt file



```
101|1|Veerappan|50|Indian|Kidnap|Bad|8-8-20$
*01|2|Osama|60|Indian|Murder|Good|7-7-18$
```

Fig 12: Inmate txt file

CHAPTER-5

CONCLUSION

The RRN technique is used for Prison Management System. The Relative Record Number identifies which position in a file the record is in.

- The purpose of conducting the study and doing the project is to know How the Prison information is maintained.
- The study of data reduces the response time.

BIBLIOGRAPHY

1. Michael j. Folk, Bill Zoellick, Greg Riccardi: File structures-An object oriented approach with C++, 3rd Edition, pearson Education, 1998.

2. K.R. Venugopal, K.G. Srinivas, P.M. Krishnaraj; File Structures Using C++, Tata McGraw-Hill, 2018.

3. Scot Robert Ladd: C++ Components and Algorithms, BPB Publications, 1993.

4. Wikipedia.com.