ATTENDENCE MANAGEMENT SYSTEM

PROGRAMMING IN C

PROJECT REPORT

ABSTRACT

Attendance management system deals with the maintenance of the student's attendance details. It generates the attendance of the on the basis of presence in class. It is maintained on the daily basis of their attendance. The staffs handling the particular subject is responsible to take the attendance for all students. Only if the student is present in that particular period, the attendance will be calculated.

An accurate report based on the student attendance percentage is generated here. This system will also help in evaluating attendance eligibility criteria of a student depending on the percentage calculated.

INTRODUCTION

Attendance management system is based on a concept to maintain students record. Here user can add Student's details. This system makes easy for storing records with in a short period of time rather than maintaining records in copies. User has to input student name and prn no. The whole project is designed in 'C' Programming language and different variables and strings have been used for the development of this project. This project is easy to operate and understand by the users.

Nowadays These types of software's are used in most of the engineering colleges. Enter the daily attendance of student. It calculates overall percentage of the student.
A computer can work more efficiently than compared to a human being. The thus becomes easy for employees and hence some amount of human resource is saved. This also saves a lot of time.
In the beginning the user is provided with options such as- Feed fresh data, take attendance, check attendance percentage etc.

Once the data such as the number of students, the name and the PRN number of the students is fed into the system then using other options like take attendance and check attendance percentage the user can mark a student absent or present and calculate the attendance percentage by entering his/her PRN number. The 4^{th} option is to exit the system when the attendance logging procedure is completed.

REQUIREMENTS:

- Structures
- File handling functions
- Loops

FUNCTIONS:

1. **fopen():**The **fopen() method** in C is a library function that is used to open a file to perform various operations which include reading, writing etc. along with various modes. If the file exists then the particular file is opened else a new file is created.

Syntax: *fopen(const char *file_name, const char *mode_of_operation);

2. **fwrite():**The fwrite() function writes the data specified by the void pointer ptr to the file.

Syntax: size_t fwrite(const void *ptr, size_t size, size_t n, FILE *fp);

ptr: it points to the block of memory which contains the data items to be written.
size: It specifies the number of bytes of each item to be written.
n: It is the number of items to be written.
fp: It is a pointer to the file where data items will be written.
On success, it returns the count of the number of items successfully written to the file. On error, it returns a number less than n. Notice that two arguments (size and n) and return value of fwrite() are of type size_twhich on the most system is unsigned int.

3. **fread():** fread() function is the complementary of fwrite() function. fread() function is commonly used to read binary data. It accepts the same arguments as fwrite() function does.

Syntax: size_t fread(void *ptr, size_t size, size_t n, FILE *fp);
The ptr is the starting address of the memory block where data will be stored after reading from the file. The function reads n items from the file where each item occupies the number of bytes specified in the second argument. On success, it reads n items from the file and returns n. On error or end of the file, it returns a number less than n.

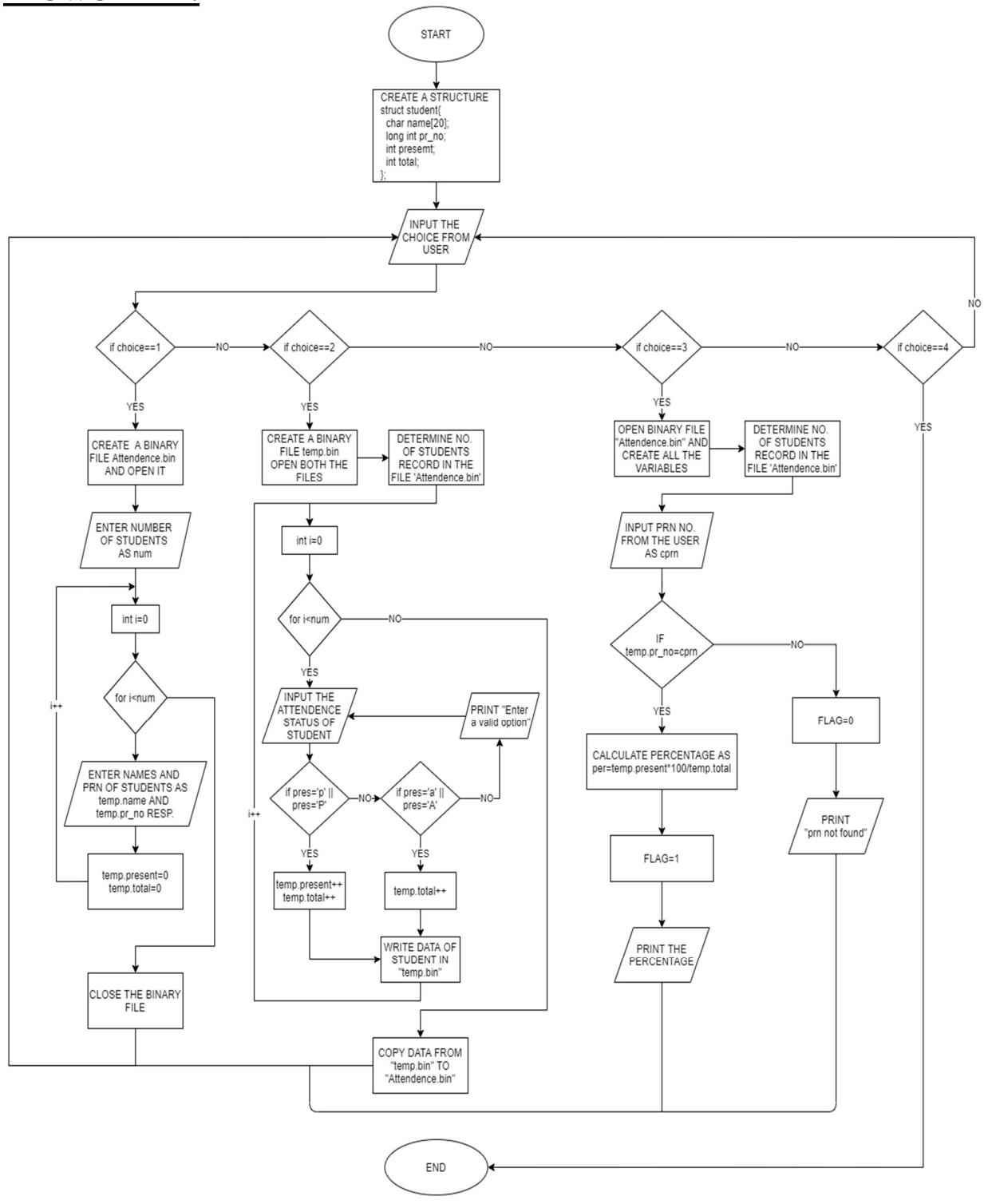
4. **fclose():**fclose() function is C library function and it's used to releases the memory stream, opened by fopen() function.

Syntax: int fclose(FILE * stream);

ALGORITHM:

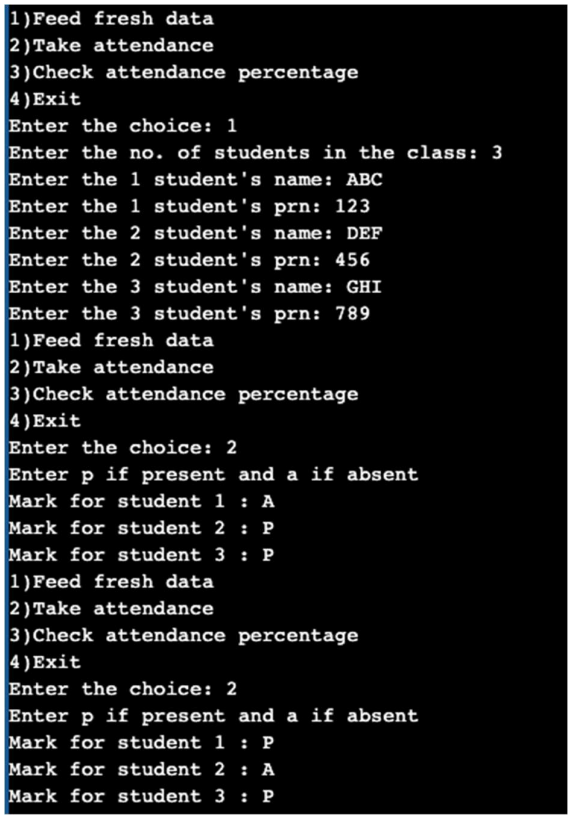
Step1. Start
Step2. Make a structure student with char name, long int pr_no,int present(for total no. of days the student was present),int total (total no. of days the attendance has been taken)
Step3. Print the menu
Step4. If input option is 1 then create a binary file “attendance.bin”
Step5. Open the file “Attendance.bin”
Step6. Input the no. of students in the class
Step7. Take the details of all the students
Step8. Write the input data in the binary file
Step9. Close the binary files
Step10. If the input option is 2 then create a binary file “temp.bin
Step11. Open the files “Attendance.bin” and “temp.bin”
Step12. determine the no. of students records in the file “attendance”
Step13. Take the data out from the file “attendance.bin”
Step14. Input if the student is present or absent
Step15. If the student is present then add 1 to the present counter and the total counter
Step16. If the student is absent then add 1 to the total counter
Step17. Write the data of the student in the file “temp.bin”
Step18. Copy the data from the file “temp.bin” to “Attendance.bin”
Step19. If the input option is 3 then open the binary file “Attendance.bin”
Step20. Determine the no. of students records in the file “Attendance.bin”
Step21. Input the pr no. of the student
Step22. Input the data from the file in a loop
Step23. Check if the input pr_no. and the pr_no. of the file matches
Step24. If it matches then calculate the attendance percentage by the formula (present counter)*100/(total counter) and put the value of flag=1 and break the loop
Step25. If the input pr_no. does not match any of the records the put the value of flag=0
Step26. Check the value of flag
Step27. If the value of flag=1 then print all the details of the required students and their attendance percentage
Step28. If the value of flag=0 then print “prn not found”
Step29. Input the choice again until unless input option is 4
Step30. If input option is 4 then quit the program
Step31. Stop

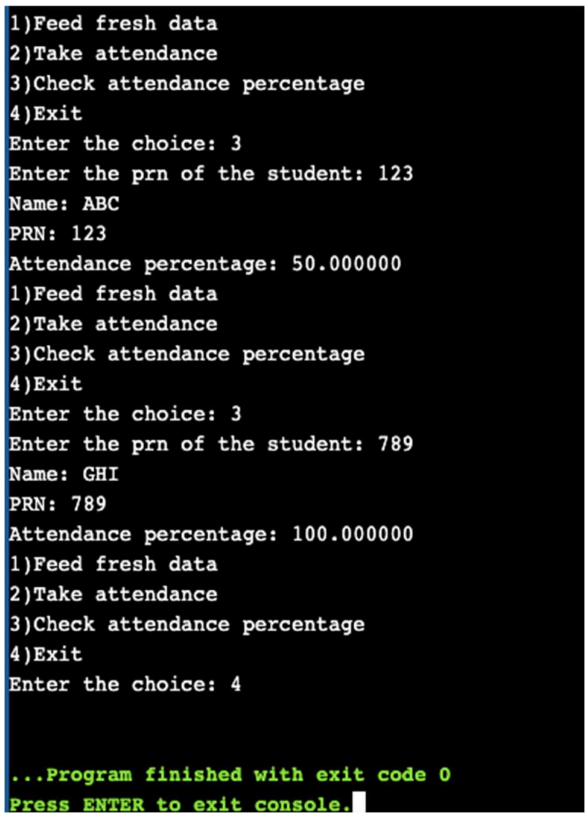
FLOWCHART:


START
CREATE A STRUCTURE
struct student{
char name[20];
long int pr_no;
int presemt;
int total;
};
INPUT THE CHOICE FROM USER
if choice==1
NO
if choice==2
NO
if choice==3
NO
if choice==4
NO
YES
YES
YES
YES
CREATE A BINARY FILE Attendence.bin AND OPEN IT
ENTER NUMBER OF STUDENTS AS num
int i=0
for i<num
i++
ENTER NAMES AND PRN OF STUDENTS AS temp.name AND temp.pr_no RESP.
temp.present=0
temp.total=0
CLOSE THE BINARY FILE
CREATE A BINARY FILE temp.bin OPEN BOTH THE FILES
DETERMINE NO. OF STUDENTS RECORD IN THE FILE 'Attendence.bin'
int i=0
for i<num
NO
YES
INPUT THE ATTENDENCE STATUS OF STUDENT
PRINT "Enter a valid option"
if pres='p' || pres='P'
NO
if pres='a' || pres='A'
NO
YES
YES
temp.present++
temp.total++
temp.total++
WRITE DATA OF STUDENT IN "temp.bin"
i++
COPY DATA FROM "temp.bin" TO "Attendence.bin"
OPEN BINARY FILE "Attendence.bin" AND CREATE ALL THE VARIABLES
DETERMINE NO. OF STUDENTS RECORD IN THE FILE 'Attendence.bin'
INPUT PRN NO. FROM THE USER AS cprn
IF temp.pr_no=cprn
NO
YES
CALCULATE PERCENTAGE AS per=temp.present*100/temp.total
FLAG=1
PRINT THE PERCENTAGE
FLAG=0
PRINT "prn not found"
END

}

SNAPSHOTS OF OUTPUT:





CONCLUSION:

By doing this project we got the opportunity to improve our coding skills further and learn new things. We used various commands such as structures, file handling functions, strings, variables, loops, etc. for creating this code on Attendance logging system. This project is efficient and simple to use and provides the user, specially staff members easy access to the system for taking attendance and calculating attendance percentage without wasting time in doing all this manually by saving lots of time.