|  |  |
| --- | --- |
| **Ex.No:15.b** | **FILE ALLOCATION STRATEGIES** |
| **LINKED** |

# AIM:

To write a C program for random access file for processing the employee details.

# ALGORITHM:

Step-1: Start the program.

Step-2: Get the number of records user want to store in the system.

Step-3: Using Standard Library function open the file to write the data into the file. Step-4: Store the entered information in the system.

Step-5: Using do..While statement and switch case to create the options such as 1-DISPLAY, 2.SEARCH, 3.EXIT.

Step-6: Close the file using fclose() function. Step-7: Process it and display the result.

Step-8: Stop the program.

# PROGRAM:

#include<stdio.h> #include<conio.h> #include<string.h> struct record

{

char empname[20]; int age;

float salary;

};

typedef struct record person FILE \*people;

void main()

{

person employee; int I,n;

FILE \*fp;

printf(“How many records:”); scanf(“%d”,&n); fp=fopen(“PEOPLE.txt”,”w”); for(i=0;i<n;i++)

{

printf(“Enter the employee information :%d(EmpName, Age,Salary):”,i+1); scanf(“%s%d%f”,employee.empname,&employee.age,& employee.salary);

fwrite(,&employee.sizeof(employee,1,people);

}

fclose(fp); int rec,result;

people=fopen(“PEOPLE.txt”,”r”);

printf(“Which record do you want to read from file?); scanf(“%d”,&rec);

while(rec>=0)

{

fseek(people,rec\*sizeof(employee),SEEK\_SET); result=fread(&em[ployee,sizeof(employee),1,people) if(result==1)

{

printf(“\n RECORD %d\n”,rec);

printf(“Given name:%s\n”, employee.empname); printf(“Age:%d years\n”,employee.age); printf(“Current salary:$ %8.2f\n\n”,employee.salary);

}

else

printf( “\n RECORD %d not found !\n\n”,rec); printf(“Which record do you want(0to3)”): scanf(“%d”<&rec):

}

fclose(people): getch():

}