

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <conio.h>
4
5  struct employee{
6      char id[10];
7      char name[20];
8      char surname[20];
9      char position[20];
10     float salary;
11 }emp;
12
13 void insert_data();
14 void delete_data();
15 void search_data();

```

บรรทัดที่ 1 – 3 include

บรรทัดที่ 5 – 11 สร้าง structure

บรรทัดที่ 13 – 15 สร้างฟังก์ชันขึ้นมารอไว้

```

17  int main(){
18      int choice;
19      char tmp[20];
20      do{
21          system("cls");
22          printf("#####\n");
23          printf("###  Welcome to employee system  ###\n");
24          printf("#####\n");
25          printf("###      Please select choice      ###\n");
26          printf("###          1. Insert Data          ###\n");
27          printf("###          2. Delete Data          ###\n");
28          printf("###          3. Search Data          ###\n");
29          printf("###          4. Exit                  ###\n");
30          printf("#####\n");
31          printf("Please select choice(1-4 only): ");
32          gets(tmp);
33          choice = atoi(tmp);
34          if(choice==1)insert_data();
35          else if(choice==2)delete_data();
36          else if(choice==3)search_data();
37      }while(choice!=4);
38      system("cls");
39      printf("#####\n");
40      printf("##  End of the program!  ##\n");
41      printf("#####\n");
42      return 0;
43  }

```

ส่วนของฟังก์ชัน main

บรรทัดที่ 20 – 37 เป็นเมนูสำหรับให้เลือกว่าต้องการทำงานอะไรกับระบบ

หากเลือกเมนูที่ 1 คือ Insert Data(เพิ่มข้อมูล) ก็จะเข้าสู่ฟังก์ชัน insert_data()

หากเลือกเมนูที่ 2 คือ Delete Data(ลบข้อมูล) ก็จะเข้าสู่ฟังก์ชัน delete_data()

หากเลือกเมนูที่ 3 คือ Search Data(ค้นหาข้อมูล) ก็จะเข้าสู่ฟังก์ชัน search_data()

หากเลือกเมนูที่ 4 ก็จะจบการทำงานของโปรแกรม โปรแกรมจะวนลูปทำงานไปเรื่อยๆจนกว่าผู้ใช้งานจะเลือกเมนูที่ 4 เพื่อออกโปรแกรม

```

45 void insert_data(){
46     FILE *fp;
47     char ans;
48     char tmp[20];
49
50     if((fp=fopen("employee.txt","ab"))==NULL){
51         printf("Can't open file for insert data\n");
52         getch();
53         exit(1);
54     }
55     do{
56         printf("\nPlease insert employee data\n");
57         printf("Employee ID : "); gets(emp.id);
58         printf("Name      : "); gets(emp.name);
59         printf("Surname   : "); gets(emp.surname);
60         printf("Position  : "); gets(emp.position);
61         printf("Salary    : "); gets(tmp);
62         emp.salary = atof(tmp);
63         fwrite(&emp,sizeof(emp),1,fp);
64         if(ferror(fp)){
65             printf("Error for insert data\n");
66             getch();
67             exit(1);
68         }
69         printf("Continue(press C) or Exit(press E) : ");
70         ans = getche();
71         while(ans!='C'){
72             if(ans=='E'){
73                 break;
74             }else if(ans!='C'){
75                 system("cls");
76                 printf("\nPlease enter only C and E\n");
77                 printf("Continue(press C) or Exit(press E) : ");
78                 ans = getche();
79             }
80         }
81     }while(ans!='E');
82     fclose(fp);
83 }

```

บรรทัดที่ 45 – 83 เป็นการทำงานของฟังก์ชัน `insert_data()` เพื่อทำการเพิ่มข้อมูลพนักงานเข้าไปยังไฟล์ `.txt`

บรรทัดที่ 50 – 54 เปิดไฟล์ `employee.txt` ในโหมด `ab` เพื่อเขียนข้อมูลต่อท้ายข้อมูลเดิมในไฟล์

บรรทัดที่ 55 – 81 วนลูปรับค่าพนักงานแต่ละคนเพื่อนำไปเขียนลงในไฟล์ จะรับข้อมูลคนต่อไปถ้าหากผู้ใช้งานป้อนค่า 'C' และจะจบการทำงานแล้วกลับไปหน้าแรกถ้าป้อนค่า 'E' ถ้าป้อนค่านอกเหนือจาก 2 ตัวนี้จะวนลูปจนกว่าจะป้อนค่าที่ถูกต้องดังในบรรทัดที่ 69 – 80

บรรทัดที่ 56 – 62 จะรับข้อมูล `id,name,surname,position,salary` ของพนักงานจากคีย์บอร์ดเก็บลงในตัวแปร `emp` ที่อ้างอิง `structure employee`

บรรทัดที่ 63 ทำการเขียนข้อมูลที่เก็บอยู่ใน `structure employee` ลงไฟล์

บรรทัดที่ 64 – 68 ตรวจสอบด้วยฟังก์ชัน `ferror()` ว่าสามารถเขียนลงไฟล์ได้สำเร็จหรือไม่

```

85 void delete_data(){
86     FILE *fp;
87     char ans;
88     char rec[10];
89     if((fp=fopen("employee.txt","rb+"))==NULL){
90         printf("Can't open file for delete data\n");
91         getch();
92         exit(1);
93     }
94     printf("\nThis is employee for delete\n");
95     do{
96         int i=0;
97         while(1){
98             fread(&emp,sizeof(emp),1,fp);
99             if(ferror(fp)){
100                 printf("\nError to read data for delete\n");
101                 exit(1);
102             }
103             if(feof(fp)){
104                 break;
105             }
106             ++i;
107             printf("%d.%s\t\t",i,emp.id);
108         }
109         printf("\nDo you want to delete data(Y or N) : ");
110         ans = getche();
111         if(ans=='N'){
112             break;
113         }

```

```

114     }
115     else if(ans=='Y'){
116         printf("\nEnter record number : ");
117         gets(rec);
118         fseek(fp,((atoi(rec)-1)*sizeof(emp)),SEEK_SET);
119         strcpy(emp.id,"\0");
120         fwrite(&emp,sizeof(emp),1,fp);
121         if(ferror(fp)){
122             printf("Error for delete data\n");
123             getch();
124             exit(1);
125         }
126         printf("Delete data complete!\n");
127         printf("\nPlease press any button to return to the home page.");
128         getch();
129         break;
130     }
131     else{
132         printf("\nPlease enter only Y and N\n");
133     }
134 }while(1);
135 fclose(fp);
}

```

บรรทัดที่ 85 – 135 เป็นการทำงานของฟังก์ชัน `delete_data()` เพื่อลบข้อมูลพนักงานออกจากไฟล์

บรรทัดที่ 89 – 93 เปิดไฟล์ `employee.txt` ในโหมด `rb+` เพื่ออ่านหรือเขียนข้อมูลลงไฟล์

บรรทัดที่ 95 – 133 ววนลูปเพื่อลบข้อมูลพนักงานออกจากไฟล์ โดยบรรทัดที่ 97 – 108 ววนลูปแสดงรหัสพนักงานทุกคน

บรรทัดที่ 98 ทำการอ่านข้อมูลจากไฟล์ บรรทัดที่ 99 - 102 ตรวจสอบว่าสามารถอ่านข้อมูลจากไฟล์ได้สำเร็จหรือไม่

บรรทัดที่ 103 – 105 ฟังก์ชัน `feof()` ตรวจสอบว่าทำการอ่านข้อมูลจากไฟล์จบแล้วหรือยังถ้าจบแล้วจะใช้คำสั่ง `break` และออกจากลูปการทำงาน ในบรรทัดที่ 97-108 ถ้ายังก็จะแสดงรหัสพนักงานต่อ

บรรทัดที่ 109 – 110 เป็นการสอบถามว่าต้องการลบข้อมูลพนักงานหรือไม่ หากต้องการ('Y') ก็จะเข้าสู่การทำงาน

ในบรรทัดที่ 114 – 129 ซึ่งบรรทัดที่ 115-116 เป็นการกรอกเลขลำดับของรหัสพนักงานที่ต้องการลบข้อมูล บรรทัดที่ 117 จะใช้ฟังก์ชัน `fseek()` เพื่อเลื่อน `file pointer` ไปยังเลขที่เรากรอก แล้วทำการเซตค่า `id` เป็น `NULL (\0)`

ในบรรทัดที่ 118 เท่ากับว่าลบค่าข้อมูลนั้นไปแล้ว

ถ้าไม่ต้องการ('N') ก็จะเข้าสู่การทำงานในบรรทัดที่ 111 – 113 ถ้าป้อนค่านอกเหนือจากนั้นก็จะเข้าสู่การทำงานในบรรทัดที่ 130 – 132 เพื่อแสดงข้อความให้กรอกเฉพาะ Y และ N และวนลูปการทำงานอีกครั้ง

```

137 void search_data(){
138     FILE *fp;
139     char ans;
140     char id[10];
141     if((fp=fopen("employee.txt","rb"))==NULL){
142         printf("Can't open file for search data\n");
143         getch();
144         exit(1);
145     }
146     do{
147         printf("\nPlease enter employee id for search : ");
148         scanf("%s",id);
149
150         while(1){
151             fread(&emp,sizeof(emp),1,fp);
152             if(ferror(fp)){
153                 printf("Error for search data\n");
154                 getch();
155                 exit(1);
156             }
157             if(feof(fp)){
158                 break;
159             }
160             if(strcmp(id,emp.id)==0){
161                 printf("Data Found!!\n");
162                 printf("Employee ID : %s\n",emp.id);
163                 printf("Name      : %s\n",emp.name);
164                 printf("Surname   : %s\n",emp.surname);
165                 printf("Position  : %s\n",emp.position);
166                 printf("Salary    : %.2f\n\n",emp.salary);
167                 break;
168             }
169         }
170         if(strcmp(id,emp.id)!=0){
171             printf("No employee in employee id %s\n\n",id);
172         }

```

```

173         printf("Continue(press C) or Exit(press E) : ");
174         ans = getche();
175         rewind(fp);
176         while(ans!='C'){
177             if(ans=='E'){
178                 break;
179             }else if(ans!='C'){
180                 system("cls");
181                 printf("\nPlease enter only C and E\n");
182                 printf("Continue(press C) or Exit(press E) : ");
183                 ans = getche();
184             }
185         }
186         printf("\n");
187     }while(ans!='E');
188     fclose(fp);
189 }

```

บรรทัดที่ 137 – 189 การทำงานของฟังก์ชัน `search_data()` เพื่อค้นหาข้อมูลของพนักงาน

บรรทัดที่ 141 – 145 เปิดไฟล์ `employee.txt` ในโหมด `rb` เพื่ออ่านข้อมูล

บรรทัดที่ 147 – 148 รับค่า `id` เข้ามาจากผู้ใช้งาน เพื่อค้นหาข้อมูลของพนักงานคนนั้น

บรรทัดที่ 151 – 155 ทำการอ่านข้อมูลจากไฟล์

บรรทัดที่ 157 – 159 ตรวจสอบว่าอ่านข้อมูลจบไฟล์หรือยัง ถ้าจบไฟล์แล้วให้ออกจากลูปการทำงานในบรรทัดที่ 150 – 169

บรรทัดที่ 160 – 168 เปรียบเทียบข้อมูล `id` ที่ได้จากไฟล์ว่ามีค่าตรงกับ `id` ที่ผู้ใช้ค้นหาหรือไม่ ถ้าตรงกันจะแสดงข้อมูลของพนักงานคนนั้นออกทางจอภาพ เมื่อแสดงผลเสร็จแล้วจะใช้คำสั่ง `break` เพื่อออกจากลูปในบรรทัดที่ 150 – 169

บรรทัดที่ 170 – 172 หากอ่านข้อมูลจนจบไฟล์แล้ว ข้อมูลของ `id` ไม่ตรงกับ `id` ที่ค้นหา ก็จะแสดงข้อความว่าไม่พบข้อมูล

บรรทัดที่ 173 – 174 ถามผู้ใช้งานว่าต้องการค้นหาข้อมูลอีกครั้งหรือไม่ ถ้าไม่ ก็จะออกจากโปรแกรมกลับไปหน้าแรก ถ้าต้องการก็จะวนลูปกลับไปค้นหาข้อมูลอีก ถ้าป้อนค่านอกเหนือจาก `'C'` และ `'E'` ก็จะวนลูปรับค่าจนกว่าจะกรอกถูก

บรรทัดที่ 175 ฟังก์ชัน `rewind()` เพื่อเลื่อน `file pointer` ไปยังต้นไฟล์ เพื่อเตรียมสำหรับการค้นหาข้อมูลของคนต่อไป