

```

1  /*#####*/
2  /* HW09_HASAN_MEN_131044009_part3.c */
3  /* HAZIRLAYAN : HASAN MEN - 131044009 */
4  /* TARİH: 27.4.15 */
5  /* */
6  /* TANIM: text dosyasından bilgileri alıp binary dsysına yazan, */
7  /* daha sonra binary dosyasından okuma yaparak tablolara dolduran ve */
8  /* istenildiği takdirde tablolardaki elemanlardan arama yapip sonucu */
9  /* ekrana basan program */
10 /*#####*/
11 #include <stdio.h>
12
13 #define TEXT_FILE "input3.txt"
14 #define BIN_FILE "output1.bin"
15
16 #define MAX_SIZE 100 /* mpn_table için maximum boyut*/
17
18
19 typedef struct /* kombinasyonların tutulacağı structlar */
20 {
21     int first,second,third;
22 }triplet_t;
23
24 typedef struct /* tüm bilgilerin atılması için bolumler */
25 {
26     int mpn,lower,upper;
27     triplet_t combination;
28 }Row;
29
30 /* input parametre olarak aldığı text ve bin dosyasları icinden texten okuma
31 yapip binarye e yazar . yazma islemi 10'lu olarak yapılmıştır */
32 void Into_Binary(FILE *text_input,FILE *binary_input);
33
34 /* binary filedan okuma yaparak 10lu olarak struct tablomuzu doldurur ve kac
35 deger doldurulduysa return edeilir */
36 int Load_Mpn_table(FILE *binary_input_file,Row mpn_table[],int maxsize);
37
38 /* kullanıcidan aldığı kombinasyon degerlerine gore tablodaki arama yapip
39 sonuclari terminale basar */
40 void search (Row mpn_table[],int actual_size, const triplet_t triplet_to_search);
41
42 int main()
43 {
44     Row table_mpn[MAX_SIZE]; /* binaryden okunacaklarla doldurulacak */
45     triplet_t search_comb; /* kullanıcidan alınacak arama degerleri */
46     int new_table_size; /* dosyadan okunan bilgi adedi */
47
48     FILE *text,*bin; /* text ve binary dosyalarımız acildi */
49     text=fopen(TEXT_FILE,"r");
50     bin=fopen(BIN_FILE,"wb"); /*write modunda */
51
52     /* binaryye bilgiler yazildi ve read modu kapatilip read binary acildi */
53     Into_Binary(text,bin);
54     fclose(text);
55     fclose(bin);
56     bin=fopen(BIN_FILE,"rb");
57
58     /* binaryden bilgiler tabloya dolduruldu */
59     new_table_size = Load_Mpn_table(bin,table_mpn,MAX_SIZE);
60     /*printf("Upper2 = %d // size = %d\n",table_mpn[1].upper,new_table_size);*/
61
62     /* kullanıcı ilk degeri -1 girene kadar kombinasyonlar icinde arama yapar*/
63     do
64     {
65         printf("\nEnter a combination-of-positives triptlet\n");
66         printf("Enter -1 x x to exit!!!(x some ingeter)\n");
67         scanf("%d%d%d", &(search_comb.first),
68             &(search_comb.second),
69             &(search_comb.third));
70
71         search(table_mpn,new_table_size,search_comb);
72

```

```

73     }while((search_comb.first)!=-1);
74
75     /* binary dosyasi kapatildi ve program sonlandi */
76     fclose(bin);
77     return 0;
78 }
79
80 /* text dosyasindan bilgileri binarye gecirir */
81 void Into_Binary(FILE *text_input,FILE *binary_input)
82 {
83     int i=0;
84     Row temp[20];
85     Row bin_temp[20];
86     char ctemp;
87
88     /* EOF gorene kadar bilgileri okur ve kac tane okundugunu tutar.*/
89     while(fscanf(text_input,"%d %c%d %c%d %d %d %d",
90                 &(temp[i].combination.first),
91                 &ctemp,
92                 &(temp[i].combination.second),
93                 &ctemp,
94                 &(temp[i].combination.third),
95                 &(temp[i].mpn),
96                 &(temp[i].lower),
97                 &(temp[i].upper))!=EOF)
98     {
99         i++;
100    }
101
102    printf("%d - Record saved to binary file\n",i);
103    /* okunan bilgiler binary dosyasina yazildi */
104    fwrite(&temp,sizeof(Row),i,binary_input);
105
106    /* kontrol icin binary den okuma yapildi */
107    /*
108    fclose(binary_input);
109    binary_input=fopen(BIN_FILE,"rb");
110
111    fread(&bin_temp,sizeof(Row),i,binary_input);
112    printf("Bin_upper2 = %d\n",bin_temp[1].upper);*/
113
114 }
115
116 /* binary_input'tan okuma yapilir ve kac tane bilgi okunduysa dondurulur */
117 /* mpn_table output digerleri input parametre olarak alinmistir */
118 int Load_Mpn_table(FILE *binary_input_file,Row mpn_table[],int maxsize)
119 {
120     int read=0,check;
121     /* kac bilgi okundugunu tutar */
122     check = fread(&mpn_table[0],sizeof(Row),10,binary_input_file);
123
124     /* tutulan bilgi varsa devam */
125     /* 10lu sekilde okuma yapar */
126     while(check!=0)
127     {
128         read+=check; /* toplam okunan bilgileri topla */
129         /* bilgi okumasi bitene yani eof olana kadar devamm eder */
130         check = fread(&mpn_table[read],sizeof(Row),10,binary_input_file);
131     }
132     printf("%d - Record readed from binary file\n",read);
133     return read; /* toplam okunan bilgi return edildi */
134 }
135
136 /* tum degiskenler input parametre olarak kullanildi */
137 /* triplet_t turunde kullanicidan alinan degerleri mpn_Table icinde ariyip */
138 /* sonucu ekrana basar */
139 void search (Row mpn_table[],int actual_size, const triplet_t triplet_to_search)
140 {
141     int i=0,found=0;
142
143     /* kac eleman icinde aranacak */
144     while(i<actual_size)

```

```
145     {
146         if(mpn_table[i].combination.first==triplet_to_search.first &&
147            mpn_table[i].combination.second==triplet_to_search.second &&
148            mpn_table[i].combination.third==triplet_to_search.third)
149         {
150             /* bulunursa bilgilendir ve bulunduğunu isaret et */
151             printf("MPN = %d;%d%% of samples contaion between %d and %d bacteria/ml \
152             \n#####\n",
153                 mpn_table[i].mpn,95,mpn_table[i].lower,mpn_table[i].upper);
154             found=1;
155         }
156         /* sonraki elemanda arama yap */
157         i++;
158     }
159     /* bulunmaz ise bilgi ver */
160     if(!found)
161         printf("Combination couldn't found.Try again \
162         \n#####\n");
163 }
164
165 /* HW09_HASAN_MEN_131044009_part3.c sonu */
```