

1.2.1. Prelucrarea bazelor de date. Indicații generale.

Condițiile din fiecare variantă includ câte o bază de date, care conține informație specifică pentru fiecare variantă în parte. Rezolvarea problemei se reduce la prelucrarea bazei de date conform condițiilor variantei, care include organizarea căutării în bază după un câmp oarecare, ordonarea înscrierilor, alte operații.

Primul pas în rezolvarea sarcinii va fi afișarea condițiilor problemei la monitor în mod program. Apoi utilizatorul va primi invitația de a indica numărul dorit de înscrieri din care va fi compusă baza. Completarea bazei de date, adică atribuirea valorilor fiecărui câmp din înscrieri, va fi făcută în mod manual de la tastatură. După aceasta baza de date va fi înscrisă într-un fișier cu nume "baza.txt" pentru ca mai apoi să fie citită din el, afișată și prelucrată.

După completarea bazei de date și înscrierea ei în fișierul "baza.txt", aceasta va fi citită din fișier și afișată la monitor într-o formă comodă de citit, adică toată informația afișată va fi amplasată într-un tabel care va conține un antet cu denumirea câmpurilor și un număr concret de înscrieri. Mai mult ca atât, baza de date va fi afișată în câteva culori, în așa fel ca două înscrieri vecine să fie de culori diferite.

După afișarea bazei utilizatorul va primi invitația de a mai introduce informație într-o înscriere. Dacă invitația va fi acceptată, această înscriere va fi înscrisă direct în fișierul "baza.txt" și amplasată la sfârșitul acestuia, adică la sfârșitul bazei de date. După introducerea acestei informații invitația va fi repetată. Apoi baza completată va fi din nou afișată la monitor.

Apoi urmează rezolvarea tuturor condițiilor din problemă. Pentru acest scop vor fi create câteva subprograme, fiecare din ele rezolvând o subsarcină concretă. Deasemenea pentru acțiunile ce au fost descrise mai sus, adică completarea bazei, înscrierea în fișier și afișarea ei deasemenea pot fi folosite subprograme aparte. Cantitatea de subprograme va depinde de condițiile problemei și dorința executorului lucrării de curs.

După rezolvarea sarcinii și afișarea rezultatelor primite baza de date finală va fi din nou afișată la monitor într-o formă comodă de citit.

Dacă sarcina lucrării de curs presupune aranjarea înscrierilor într-un mod sau altul, atunci baza finală cu înscrierile aranjate va fi înscrisă în alt fișier cu nume "bazanoua.txt".

Cu scopul de a face mai comodă utilizarea programului, în cadrul lui va fi creat un meniu cu ajutorul cărui va fi simplu de trecut dintr-o etapă a rezolvării în alta. În așa fel programul trebuie să conțină următoarele puncte de meniu:

1. Afișarea condițiilor problemei.

2. Completarea bazei de date și înscrierea ei în fișierul “baza.txt”.
3. Citirea bazei de date din fișierul “baza.txt” și afișarea ei la monitor.
4. Întroducerea unei noi înscrieri și afișarea bazei noi la monitor.
5. Căutare în baza de date.
6. Alte condiții conform variantei.
7. Aranjarea bazei de date și înscrierea ei în fișierul “bazanoua.txt”.
8. Afișarea bazei finale ordonate.
9. Ieșire

După îndeplinirea oricărui punct din meniu programul va reveni din nou la afișarea meniului. După rezolvarea fiecărui punct rezultatul corespunzător va fi afișat la monitor. Punctele din meniu 5,6,7 ce corespund rezolvării sarcinilor conform variantelor pot fi unite într-un număr mai mic sau despărțite într-un număr mai mare de puncte de meniu în dependență de condițiile problemei și dorința executorului lucrării de curs.

1.2.2. Exemplu de rezolvare a unei sarcini

De alcătuit o bază de date cu N înscrieri care ar conține informația despre calculatoarele într-un laborator de programare.

- a) De organizat căutările calculatoarelor după nume și sistem operațional.
- b) De aflat calculatoarele cu memoria ROM în intervalul propus de utilizator.
- c) De aranjat înscrierile în ordine descrescătoare după memoria RAM.

Listingul programului:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#include<stdlib.h>
#include<string.h>
//variabile globale
int z=0;
struct pc {
char nume[15],so[22];
float rom,ram;}aux;
struct pc x[50];

//afisarea conditiilor initiale
void conditii (void) {
```

```

clrscr(); gotoxy(20,2);textcolor(11);
cprintf("Lucrare de curs la disciplina programare");
gotoxy(29,4);textcolor(15); cprintf("Conditiiile problemei:");
gotoxy(10,5);textcolor(11);
cprintf("De alcatuit o baza de date cu N inscrieri care ar contine informatia");
gotoxy(10,6);
cprintf("despre calculatoarele intr-un laborator de programare.");
gotoxy(5,7);
cprintf("a)De organizat cautarile calculatoarelor dupa nume si sistem operational");
gotoxy(5,8);
cprintf("b)De aflat calculatoarele cu memoria ROM in intervalul propus de
utilizator");
gotoxy(5,9);
cprintf("c)De aranjat inscrierile in ordine descrescatoare dupa memoria
RAM");gotoxy(5,wherey()+2);
printf("┌───────────────────────────────────────────────────────────────────────────────────┐");
gotoxy(5,wherey()+1);
printf("│   Nume   │ Sistemul Operational │ ROM (Gb) │ RAM (Mb) │");
gotoxy(5,wherey()+1);
printf("├──────────┤──────────────────────────┤──────────┤──────────┤");
gotoxy(5,wherey()+1);
printf("│           │           │           │           │");
gotoxy(5,wherey()+1);
printf("│           │           │           │           │");
gotoxy(5,wherey()+1);
printf("│           │           │           │           │");
gotoxy(5,wherey()+1);
printf("│           │           │           │           │");
gotoxy(5,wherey()+1);
printf("│           │           │           │           │");
gotoxy(5,wherey()+1);
printf("└──────────┤──────────────────────────┤──────────┤──────────┤");
textcolor(15);gotoxy(5,wherey()+2);
cprintf("Pentru iesire in meniu culegeti <ENTER>"); getch(); }

```

```

//completarea bazei de date
int completare(void) {

```

```

int i,k; float a,b; clrscr(); gotoxy(15,2);
printf("Culegeti cantitatea de inscrieri in baza "); scanf("%d",&k);
for(i=0;i<k;i++) { clrscr();
gotoxy(5,2); printf("Indicati numele statiei %d ",i+1); scanf("%s",x[i].nume);
gotoxy(5,wherey()+2); printf("Indicati sistemul operational de pe statia %d ",i+1);
scanf("%s",x[i].so);
gotoxy(5,wherey()+2);printf("Indicati marimea memoriei ROM (in Gb) a statiei %d
",i+1); scanf("%f",&a); x[i].rom=a;
gotoxy(5,wherey()+2);printf("Indicati marimea memoriei RAM (in Mb) a statiei %d
", i+1); scanf("%f",&b); x[i].ram=b; } textcolor(15); gotoxy(5,wherey()+2);
cprintf("Pentru iesire in meniu culegeti <ENTER>"); getch(); return(k); }

```

//inscrierea bazei in fisier

```

void scrie_fisier(int k, char numef[20]) { FILE *f; int i;
if((f=fopen(numef,"w"))==NULL) { clrscr();gotoxy(15,2);textcolor(15);
cprintf("Fisierul nu poate fi deschis"); goto exit;}
for(i=0;i<k;i++) fwrite(&x[i],sizeof(x[i]),1,f); fclose(f);
clrscr(); textcolor(15); gotoxy(5,wherey()+2);
cprintf("Baza de date a fost inscrisa in fisierul %s",numef);
exit: gotoxy(5,wherey()+1);
cprintf("Pentru iesire in meniu culegeti <ENTER>"); getch(); }

```

//citirea bazei din fisier

```

int citire_fisier(char numef[20]) { FILE *f; int i; clrscr();
if((f=fopen(numef,"r"))==NULL) {
clrscr();gotoxy(15,2);textcolor(15);
cprintf("Fisierul nu poate fi deschis"); z=1; goto exit;} i=0; z=0;
while( fread(&x[i],sizeof(x[i]),1,f)==1 ) i++; fclose(f);
exit: return(i);}

```

//afisarea bazei la monitor

```

void afisare (int k, char numef[20]) { int i;
if (z==1) {clrscr(); gotoxy(5,2); textcolor(15);
cprintf ("Baza de date n-a fost citita din fisier"); goto exit;}
else { clrscr(); gotoxy(5,2); textcolor(15);
cprintf("Baza de date citita din fisierul %s este:",numef);gotoxy(5,wherey()+2);
textcolor(15);

```

```

cprintf("┌───────────────────────────────────────────────────────────────────────────────────┐");
gotoxy(5,wherey()+1);
cprintf("┌   Nume   ┌ Sistemul Operational ┌ ROM (Gb) ┌ RAM (Mb) ┌");
gotoxy(5,wherey()+1);
cprintf("┌───────────────────────────────────────────────────────────────────────────────────┐");
gotoxy(5,wherey()+1); for(i=0;i<k;i++){
if (fmod(i,2)==0) {textbackground(12); textcolor(15);} else {textbackground(BLUE);
textcolor(10);}
cprintf("┌ %15s ┌ %22s ┌ %12.3f ┌ %12.3f ┌",x[i].nume,x[i].so,x[i].rom,x[i].ram);
gotoxy(5,wherey()+1);} textbackground(BLACK); textcolor(15);
printf("┌───────────────────────────────────────────────────────────────────────────────────┐");
} exit: gotoxy(5,wherey()+2);
cprintf("Pentru iesire in meniu culegeti <ENTER>"); getch();}

```

```

void add (char numef[20]) {
struct pc s; float a,b; FILE *f; int i; char q;
if((f=fopen(numef,"a"))==NULL) { clrscr();gotoxy(5,2);textcolor(15);
cprintf("Fisierul nu poate fi deschis"); goto exit;}
ad: clrscr();gotoxy(5,wherey()+2);textcolor(15);
cprintf("Doriti sa adaogati inca o inscriere in baza de date? y/n ");
q=getch(); if ((q=='y')||(q=='Y')) {
gotoxy(10,wherey()+2); textcolor(15);
cprintf("Indicati numele statiei "); scanf("%s",s.nume);
gotoxy(10,wherey()+1);
cprintf("Indicati sistemul operational "); scanf("%s",s.so);
gotoxy(10,wherey()+1);
cprintf("Indicati marimea memoriei ROM (in Gb) ");scanf("%f",&a); s.rom=a;
gotoxy(10,wherey()+1);
cprintf("Indicati marimea memoriei RAM (in Mb) ");scanf("%f",&b); s.ram=b;
f=fopen(numef,"a"); fwrite(&s,sizeof(s),1,f); fclose(f); goto ad;}
exit: gotoxy(5,wherey()+2); textcolor(15);
cprintf("Pentru iesire in meniu culegeti <ENTER>"); getch(); }

```

// cautare dupa nume, sistem operational si intervalul de memorie ROM

```

void cautare(int k,char caut[22]) {
int i,g; char Z[22],B[22];float jos,sus,R;
if (z==1) {clrscr(); gotoxy(5,2); textcolor(15);

```

```

cprintf("Baza de date n-a fost citita din fisier"); goto exit;}
else { clrscr(); gotoxy(5,2); textcolor(15);
if (strcmp(caut,"Nume")==0) {g=1;
cprintf("Culege numele statiei cautate "); scanf("%s",Z);}
if (strcmp(caut,"Sistemul Operational")==0) {g=2;
cprintf("Culege Sistemul Operational cautat "); scanf("%s",Z);}
if (strcmp(caut,"memoria ROM")==0) {g=3;
cprintf("Culege limita de jos a memoriei ROM (Gb) "); scanf("%f",&jos);
gotoxy(5,wherey());
cprintf("Culege limita de sus a memoriei ROM (Gb) "); scanf("%f",&sus);}
gotoxy(5,wherey()+1); textcolor(15);
cprintf("Rezultatul cautarii dupa %s este:",caut);gotoxy(5,wherey()+2);
printf("┌───────────────────────────────────────────────────────────────────────────────────┐");
gotoxy(5,wherey()+1);
printf("│      Nume      │ Sistemul Operational │ ROM (Gb) │ RAM (Mb) │");
gotoxy(5,wherey()+1);
printf("└───────────────────────────────────────────────────────────────────────────────────┘");
gotoxy(5,wherey()+1); for(i=0;i<k;i++){ textcolor(15);
if (g==1) {strcpy(B,x[i].nume); if ( strstr(strlwr(B),strlwr(Z))!=0 ) {
cprintf("│ %15s │ %22s │ %12.3f │ %12.3f │",x[i].nume,x[i].so,x[i].rom,x[i].ram);
gotoxy(5,wherey()+1);} }
if (g==2) {strcpy(B,x[i].so); if ( strstr(strlwr(B),strlwr(Z))!=0 ) {
cprintf("│ %15s │ %22s │ %12.3f │ %12.3f │",x[i].nume,x[i].so,x[i].rom,x[i].ram);
gotoxy(5,wherey()+1);} }
if (g==3) { R=x[i].rom; if ( (R>=jos)&&(R<=sus) ) {
cprintf("│ %15s │ %22s │ %12.3f │ %12.3f │",x[i].nume,x[i].so,x[i].rom,x[i].ram);
gotoxy(5,wherey()+1);} } }
printf("┌───────────────────────────────────────────────────────────────────────────────────┐");
} exit: gotoxy(5,wherey()+2);
cprintf("Pentru iesire in meniu culegeti <ENTER>"); getch();}

```

// aranjare in ordine descrescatoare dupa memoria RAM

```

void aranjare (int k) { int i,j,r; float max;
for (i=0;i<k;i++) { max=x[i].ram; r=i;
for (j=i;j<k;j++) { if (x[j].ram>max) {max=x[j].ram; r=j;} }
aux=x[i]; x[i]=x[r]; x[r]=aux;} clrscr(); gotoxy(5,2);textcolor(15);
cprintf("Baza de date a fost cicita din fisierul c:\\baza.txt");gotoxy(5,3);

```

```
cprintf("Inscrierile au fost aranjate in ordine descrescatoare dupa campul RAM (Mb)");  
gotoxy(5,5); cprintf("Pentru a vedea rezultatul accesati punctul 9 din meniu");  
gotoxy(5,6); cprintf("Pentru iesire in meniu culegeti <ENTER>");  
getch();}
```

```
void main(void) { int n,i; char w,v;  
char baza[20]="c:/baza.txt",bazanoua[20]="c:/bazanoua.txt";  
menu: clrscr();  
textcolor(15); gotoxy(15,2);  
cprintf("Alege din meniu:"); gotoxy(5,wherey()+2);  
cprintf("1: Afisarea conditiilor initiale"); gotoxy(5,wherey()+1);  
cprintf("2: Completarea bazei de date"); gotoxy(5,wherey()+1);  
cprintf(" si inscrierea ei in fisierul c:\\baza.txt"); gotoxy(5,wherey()+1);  
cprintf("3: Citirea bazei de date din fisierul c:\\baza.txt"); gotoxy(5,wherey()+1);  
cprintf(" si afisarea ei la monitor"); gotoxy(5,wherey()+1);  
cprintf("4: Adaogarea unei inscrieri in baza de date"); gotoxy(5,wherey()+1);  
cprintf("5: Cautarea calculatoarelor dupa nume"); gotoxy(5,wherey()+1);  
cprintf("6: Cautarea calculatoarelor dupa sistemul operational instalat");  
gotoxy(5,wherey()+1);  
cprintf("7: Determinarea sratiilor cu memoria ROM in intervalul propus");  
gotoxy(5,wherey()+1);  
cprintf("8: Aranjarea inscrierilor in ordine descrescatoare dupa marimea RAM");  
gotoxy(5,wherey()+1);  
cprintf(" si inscrierea bazei de date in fisierul c:\\bazanoua.txt");  
gotoxy(5,wherey()+1);  
cprintf("9: Afisarea bazei de date finale la monitor"); gotoxy(5,wherey()+1);  
cprintf("0: Iesire");  
w=getch();  
switch (w) {  
case '1': goto m1;  
case '2': goto m2;  
case '3': goto m3;  
case '4': goto m4;  
case '5': goto m5;  
case '6': goto m6;  
case '7': goto m7;
```

```
case '8': goto m8;
case '9': goto m9;
case '0': goto m0;
default : goto meniu;}
```

```
m1: conditii(); goto meniu;
m2: n=completare(); scrie_fisier(n,baza); goto meniu;
m3: n=citire_fisier(baza); afisare(n,baza); goto meniu;
m4: add(baza); goto meniu;
m5: n=citire_fisier(baza); cautare(n,"Nume"); goto meniu;
m6: n=citire_fisier(baza); cautare(n,"Sistemul Operational"); goto meniu;
m7: n=citire_fisier(baza); cautare(n,"memoria ROM"); goto meniu;
m8: n=citire_fisier(baza); aranjare(n); scrie_fisier(n,bazanoua); goto meniu;
m9: n=citire_fisier(bazanoua); afisare(n,bazanoua); goto meniu;
m0: clrscr();
gotoxy(15,2); textcolor(15);
cprintf("Intradevar doriti sa iesiti ? y/n");
v=getch(); if ((v=='n')||(v=='N')) goto meniu;
gotoxy(15,4); textcolor(15);
cprintf("Pentru iesire din program culegeti <ENTER> ");
getch(); }
```