

# **Documentatie**

## **Proiect 1**

### *1. Main:*

→ se afiseaza meniul care cuprinde 5 operatii posibile:

- 1) Adaugare tranzactii
- 2) Vizualizare tranzactii
- 3) Rapoarte financiare
- 4) Sold cont
- 5) Iesire program

→ se citeste si se valideaza optiunea aleasa de catre utilizator cu ajutorul functiilor: `operatie_aleasa_din_meniu()` si `verifica_optiune()`

```
Gestionare date financiare
Alegeti numarul operatiei pe care doriti sa o utilizati:
1. Adaugare tranzactii
2. Vizualizare tranzactii
3. Rapoarte financiare
4. Sold cont
0. Iesire program

Numar:
```

### *2. Adaugare tranzactii:*

→ apeleaza functia `optiune1()` care apeleaza functiile necesare pentru a adauga tranzactia

→ functia `indicatii_optiune1()` afiseaza un set de instructiuni pentru a informa utilizatorul de modul in care trebuie introduse datele

→ functia `date_tranzactie_noua()` gestioneaza citirea, validarea si salvarea datelor intr-o structura de date, cu ajutorul functiilor de citire si a structurii care are forma:

```
struct tranzactii
{
    char zi[5], luna[5], an[10];
    char descriere[200], tip[2]; /// tipul este venituri sau cheltuieli
    float suma;
} tranzactie[1];
```

→ functia `salveaza_tranzactie_noua()` salveaza in fisierul text : "tranzactii.txt" datele din structura de date si afiseaza un mesaj corespunzator (datele au fost salvate/ datele nu au putut fi salvate)

```
Numar: 1
Ati ales optiunea de adaugare a tranzactiilor. Introduceti datele corespunzatoare pentru a crea o noua tranzactie.
Zi tranzactie (1-31) : 2
Luna tranzactie (1-12): 3
An tranzactie: 2023
Descriere tranzactie: nu
Tip tranzactie (v pentru venituri, c pentru cheltuieli): c
Suma tranzactie: 150.7
Tranzactia a fost salvata cu succes!
```

### 3. Vizualizare tranzactii:

→ se afiseaza un submeniu pentru ca utilizatorul sa aleaga daca sa se afiseze toate tranzactiile, sau tranzactiile doar dintr-o anumita perioada

→ se incarca din fisier datele si se afiseaza pe ecran, in functie de alegerea utilizatorului, toate functiile ajutatoare fiind gestionate si apelate din functia `optiune3()`

- Se afiseaza tranzactiile din luna 11, a anului 2023

Numar: 2

1. Vizualizare tranzactii pe luna
2. Vizualizarea tuturor tranzactiilor

Introduceti numarul operatiei pe care doriti sa o efectuati: 1

Introduceti luna pentru care doriti sa vedeti tranzactiile (1-12): 11

Introduceti anul pentru care doriti sa vedeti tranzactiile: 2023

Tranzactie 1:

data: 12.11.2023

descriere: necesare

tip: cheltueli

suma: 270.9 lei

Tranzactie 2:

data: 7.11.2023

descriere: haine

tip: cheltueli

suma: 200 lei

- Se afiseaza toate tranzactiile

```
Numar: 2

1. Vizualizare tranzactii pe luna
2. Vizualizarea tuturor tranzactiilor

Introduceti numarul operatiei pe care doriti sa o efectuati: 2

Tranzactii:
Tranzactie 1:
data: 7.10.2022
descriere: salariu
tip: venituri
suma: 2500 lei

Tranzactie 2:
data: 22.10.2023
descriere: scaun
tip: cheltueli
suma: 230.5 lei

Tranzactie 3:
data: 2.3.2023
descriere: bonus
tip: venituri
suma: 400 lei

Tranzactie 4:
data: 12.11.2023
```

#### 4. Rapoarte financiare:

→ se afiseaza veniturile si cheltuielile pe o anumita perioada, citita de la tastatura

→ se citesc datele din fisier, se verifica daca tranzactia citita apartine perioadei precizate, iar daca apartine se determina daca suma reprezinta venituri sau cheltuieli si se adauga la tipul corespunzator

EXEMPLU: (perioada 2-12.2023)

```
Numar: 3

Introduceti perioada pentru care doriti sa vizualizati raportul.
Luna inceput perioada: 2

Luna sfarsit perioada: 12

An perioada: 2023

Venituri: 400.000000 lei
Cheltuieli: 1231.000000 lei
```

### 5. Sold cont:

→ se afiseaza soldul curent al contului, astfel ca se citesc tranzactiile din fisier, se determina daca suma reprezinta venituri sau cheltuieli, se adauga la tipul corespunzator, iar apoi se afiseaza diferenta dintre venituri si suma, pentru a se determina soldul curent

```
Numar: 4  
Sold curent: 1668.300049 lei
```

### 6. Iesire program

```
Numar: 0  
La revedere!
```

## 7. Operatii cu fisiere

→ Citire din fisier:

```
FILE *fptr;
fptr = fopen("D:\\UBB INFO - ANUL I (2023-2024)\\Anul I - Sem.
\\Programare in C\\Proiecte\\Proiect 1\\tranzactii.txt","r");
if(fptr==NULL)
{
    printf("\nTranzactiile nu au putut fi incarcate!\n");
    optiune2();
    return;
}
int index=1;
while(!feof(fptr))
{
    char s[1000];
    fscanf(fptr,"%s",s);
    //printf("%zu",strlen(s));
    if(strlen(s)<12)
    {
        break;
    }
    printf("Tranzactie ");
    printf("%d",index);
    printf(":\n");
    char *token=strtok(s,";");
    printf("data: ");
    printf(token);
    printf("\n");

    token=strtok(NULL,";");
    printf("descriere: ");
    printf(token);
    printf("\n");

    token=strtok(NULL,";");
    printf("tip: ");
```

```

        if(token[0]=='v')
        {
            printf("venituri");
        }
        else
        {
            printf("cheltueli");
        }
        printf("\n");

        token=strtok(NULL, ";");
        printf("suma: ");
        printf(token);
        printf(" lei\n\n");

        index++;
    }

    fclose(fptr);

```

→ Scriere in fisier:

```

FILE *fptr;
fptr = fopen("D:\\UBB INFO - ANUL I (2023-2024)\\Anul I - Sem.
\\Programare in C\\Proiecte\\Proiect 1\\tranzactii.txt", "a"); ///deschidem fisierul
pentru a adauga date la finalul acestuia
if(fptr==NULL) ///eroare de deschidere a fisierului
{
    printf("\nTranzactia nu a putut fi salvata!\n");
    optiune1();
    return;
}

fprintf(fptr, &tranzactie[0].zi);
fprintf(fptr, ".");

```



```

fprintf(fp, &tranzactie[0].luna);
fprintf(fp, ".");
fprintf(fp, &tranzactie[0].an);

fprintf(fp, ";");
fprintf(fp, &tranzactie[0].descriere);

fprintf(fp, ";");
fprintf(fp, &tranzactie[0].tip[0]);

fprintf(fp, ";");
fprintf(fp, &tranzactie[0].suma);
fprintf(fp, "\n");

fclose(fp);
printf("\nTranzactia a fost salvata cu succes!\n");

```

→ Verificare daca se poate deschide fisierul, altfel se afiseaza mesaj pentru a preveni blocarea aplicatiei

```

FILE *fp;
fp = fopen("D:\\UBB INFO - ANUL I (2023-2024)\\Anul I - Sem.
I\\Programare in C\\Proiecte\\Proiect 1\\tranzactii.txt", "a"); ///deschidem fisierul
pentru a adauga date la finalul acestuia
if(fp==NULL) ///eroare de deschidere a fisierului
{
    printf("\nTranzactia nu a putut fi salvata!\n");
    optiune1();
    return;
}

```

## 8. Operatii cu structuri

→ Declararea structurii:

```
struct tranzactii
{
    char zi[5], luna[5], an[10];
    char descriere[200], tip[2]; /// tipul este venituri sau cheltuieli
    float suma;
} tranzactie[1];
```

→ Adaugare valori in structura:

```
printf("\nDescriere tranzactie: ");
scanf("%s", &tranzactie[0].descriere);
```

```
printf("\nAn tranzactie: ");
scanf("%s", &tranzactie[0].an);
```

→ Operatii cu valorile din structura:

```
fprintf(fp_ptr, &tranzactie[0].zi);
```

```
fprintf(fp_ptr, &tranzactie[0].descriere);
```

## 9. Functii folosite pentru manipularea sirurilor de caractere

→ strtok

```
printf("Tranzactie ");
printf("%d",index);
printf(":\n");
char *token=strtok(s,";");
printf("data: ");
printf(token);
printf("\n");

token=strtok(NULL,";");
printf("descriere: ");
printf(token);
printf("\n");

token=strtok(NULL,";");
printf("tip: ");
if(token[0]=='v')
{
    printf("venituri");
}
else
{
    printf("cheltueli");
}
printf("\n");

token=strtok(NULL,";");
printf("suma: ");
printf(token);
printf(" lei\n\n");
```

→ strcmp

```
if(strcmp(luna,"1")<0 || strcmp(luna,"12")>0)
{
    return false;
}
```

→ strtod

```
char *end;
float v=strtod(suma, &end);
```

→ strcpy

```
char suma[10];
strcpy(suma,stok);
```