

Gaujaneanu Nicoleta Monica

Grupa 414D

## Examen Final Programarea Calculatoarelor

### Exercitiul 1:

Specificare formala:

#### **Date de intrare:**

numele fisierului – fisier

linia pe care vrem sa o afisam – n

sirul de caractere pentru fiecare linie – sir[50]

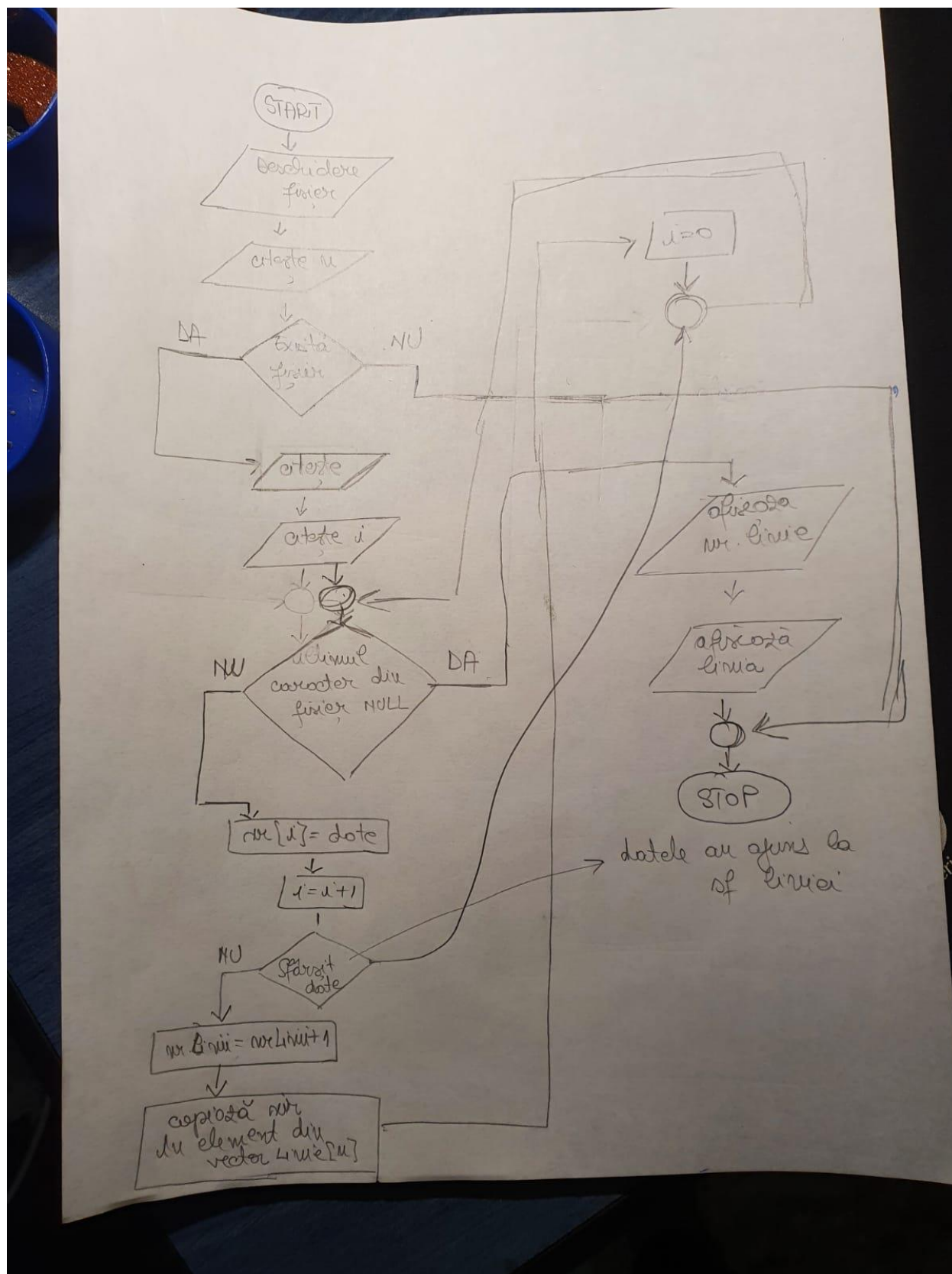
#### **Date de iesire:**

numarul de linii din fisier – nrLinii

textul de pe linia pe care vreau sa o afisez – Linie[i].text

Gaujaneanu Nicoleta Monica

Grupa 414D



Gaujaneanu Nicoleta Monica

Grupa 414D

Cod

```
#include <stdio.h>
```

```
#include <string.h>
```

```
struct Linie{    // am creat structura pentru a scrie datele intr-o inregistrare adecvat definita
```

```
    char text[50]; // am luat textul de 50 de caractere
```

```
}Linie[10]; // am luat un vector de 10 elemente, fiecare element reprezentand o linie din fisier
```

```
int main(){
```

```
    char sir[50];    // reprezinta fiecare linie din fisier
```

```
    int nrLinii = 0; // numarul total de linii si este initializat cu 0 pentru a numara liniile din fisier
```

```
    FILE *fisier = fopen("ListaGaujaneanuNicoleta.txt", "r");    // am deschis fisierul
```

```
    int n;
```

```
    printf("Dati linia pe care doriti sa o afisati: ");
```

```
    scanf("%d",&n);
```

```
    if(fisier == NULL) {    // verific daca fisierul exista, daca nu exista afiseaza mesajul ca nu se poate deschide fisierul
```

```
        perror("Nu se poate deschide fisierul!");
```

```
        exit(1);    // imi va returna valoarea 1 si imi iese din program
```

```
    }
```

Gaujaneanu Nicoleta Monica

Grupa 414D

```
char date[255];    // reprezinta caracterele din fisier. Am pus 255 deoarece lungimea fiecărei  
linii este de 23 înmulțit cu 10 = 230
```

```
int i;    // pentru a accesa un caracter din variabila sir
```

```
while(fgets(date, sizeof(date), fisier) != NULL){    // parcurg fisierul
```

```
    sir[i] = date;    // initializez variabila sir cu fiecare caracter din fisier
```

```
    i++;
```

```
    if(date == "\n"){    // cand am ajuns la sfarsitul liniei, imi va numara linia si imi va pune sirul  
    obtinut in variabila sir intr-un element din vectorul de structuri Linie
```

```
        nrLinii++;    // incrementez nr de linii
```

```
        strcpy(Linie[n].text, sir); // imi pune sirul obtinut in variabila sir intr-un element din  
vectorul de structuri Linie
```

```
        i = 0;    // reincep numararea
```

```
    }
```

```
}
```

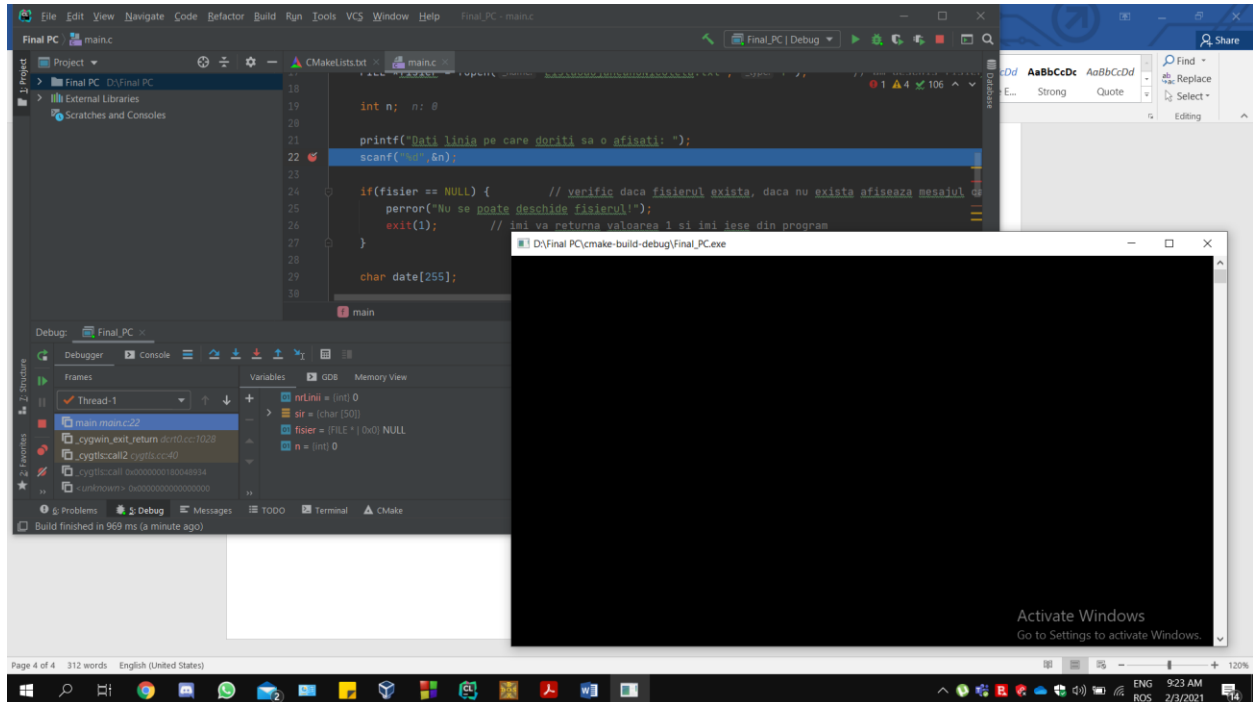
```
printf("\nNumarul de linii este: %d",nrLinii);
```

```
printf("\n Linia este: ", Linie[n].text);
```

```
fclose(fisier);
```

```
return 0;
```

```
}
```



introduc linia

## Exercitiul 2:

Specificare in limbaj natural:

Vom citi dimensiunile imaginii, si vom genera pentru fiecare element al tabloului o valoare aleatoare in intervalul  $[0, 255]$ . Vom crea regiunea de interes pe care o vom prelucra.

Apoi vom realiza, pe rand, functiile de negare a imaginii si de transformare in imagine alb-negru astfel:

Pentru functia de negare, vom da ca parametru de intrare imaginea, numarul de linii si de coloane si regiunea de interes. In aceasta functie vom parcurge pixel cu pixel zona de interes a imaginii si pentru fiecare pixel vom recalcula valoarea acestuia scazand 255 din valoarea sa. Iar apoi vom schimba valoarea pixelului initial cu valoarea nou calculata. Valorile finale vor de la 0 la 255.

Pentru functia de transformare in imagine alb-negru se vor da aceiasi parametri de intrare. In aceasta functie se va parcurge zona de interes din imagine pixel cu pixel si daca valoarea pixelului este mai mare decat 127, valoarea pixelului devine 255, iar in caz contrar de vine 0. Iar valorile obtinute le punem in pixelii din zona de interes. Valorile finale vor fi fie de 255, fie de 0.