



**Ministerul Educației și Cercetării
al Republicii Moldova**

Universitatea Tehnică a Moldovei

Departamentul Informatică și Ingineria a sistemelor

Disciplina: Programarea Calculatoarelor

RAPORT

la lucrarea de laborator nr. 3

Tema: Prelucrarea tablourilor bidimensionale în
limbajul C

A efectuat :

Veaceslav Mihalachi

Studentul gr-CR-251

A verificat :

Vladina Tutunaru

Scopul :

Studierea posibilităților și mijloacelor limbajului C pentru programarea algoritmilor de

prelucrare a tablourilor bidimensionale.

Sarcina :

Să se ordeneze liniile tabloului în ordinea ascendentă a numărului de elemente pozitive în fiecare linie.

#Programul principal 16

Input

```
1 #include <stdio.h>
2
3 int main() {
4     int t[3][3];
5     int n = 3, m = 3;
6     int count[3] = {0};
7
8     printf("introduceti numere pentru matricea 3x3:\n\n");
9     for(int i = 0; i < n; i++) {
10        for(int j = 0; j < m; j++) {
11            scanf("%d", &t[i][j]);}
12
13        for(int i = 0; i < n; i++) {
14            for(int j = 0; j < m; j++) {
15                if(t[i][j] > 0) {
16                    count[i]++;}} }
17
18        for(int i = 0; i < n-1; i++) {
19            for(int j = i+1; j < n; j++) {
20                if(count[i] > count[j]) {
21                    int temp_c = count[i];
22                    count[i] = count[j];
23                    count[j] = temp_c;
24
25                    for(int k = 0; k < m; k++) {
26                        int temp = t[i][k];
27                        t[i][k] = t[j][k];
28                        t[j][k] = temp;}}}
29        printf("tabloul ordonat:\n\n");
30        for(int i = 0; i < n; i++) {
31            for(int j = 0; j < m; j++) {
32                printf("%d ", t[i][j]);}
33                printf("\n");}
34    return 0;}
35
36
```

Output

```
introduceti numere pentru matricea 3x3:  
-1 2 5  
9 5 5  
-4 -7 3  
tabloul ordonat:  
-4 -7 3  
-1 2 5  
9 5 5
```

#Probleme pentru tablouri unidimensionale:

Input

```
main.c  
1 #include <stdio.h>  
2  
3 void citesteTablou(int arr[], int n, int i) {  
4     if(i < n) {  
5         printf("Element %d: ", i+1);  
6         scanf("%d", &arr[i]);  
7         citesteTablou(arr, n, i+1);  
8     }  
9 }  
10  
11 void afiseazaTablou(int arr[], int n, int i) {  
12     if(i < n) {  
13         printf("%d ", arr[i]);  
14         afiseazaTablou(arr, n, i+1);  
15     }  
16 }  
17  
18 int main() {  
19     int n;  
20     printf("Numar elemente: ");  
21     scanf("%d", &n);  
22  
23     int arr[n];  
24     citesteTablou(arr, n, 0);  
25  
26     printf("Tabloul: ");  
27     afiseazaTablou(arr, n, 0);  
28  
29     return 0;  
30 }
```

Output

```
Numar elemente: 3
Element 1: 2
Element 2: 4
Element 3: 1
Tabloul: 2 4 1
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9     printf("Introduceti %d elemente:\n", n);
10    for(int i = 0; i < n; i++) {
11        scanf("%d", &arr[i]);
12    }
13
14    printf("Elemente negative: ");
15    for(int i = 0; i < n; i++) {
16        if(arr[i] < 0) {
17            printf("%d ", arr[i]);
18        }
19    }
20
21    return 0;
22 }
```

Output

```
Numar elemente: 3
Introduceti 3 elemente:
-4
2
1
Elemente negative: -4
```

Input

```
main.c
1 #include <stdio.h>
2
3 int sumaRecursiv(int arr[], int n) {
4     if(n <= 0) return 0;
5     return arr[n-1] + sumaRecursiv(arr, n-1);
6 }
7
8 int main() {
9     int n;
10    printf("Numar elemente: ");
11    scanf("%d", &n);
12
13    int arr[n];
14    printf("Introduceti %d elemente:\n", n);
15    for(int i = 0; i < n; i++) {
16        scanf("%d", &arr[i]);
17    }
18
19    printf("Suma: %d\n", sumaRecursiv(arr, n));
20
21    return 0;
22 }
```

Output

```
Numar elemente: 4
Introduceti 4 elemente:
13
-2
3
7
Suma: 21
```

Input

```
main.c
1 #include <stdio.h>
2 #include <limits.h>
3
4 int maximRecursiv(int arr[], int n) {
5     if(n == 1) return arr[0];
6
7     int max_rest = maximRecursiv(arr, n-1);
8     return (arr[n-1] > max_rest) ? arr[n-1] : max_rest;
9 }
10
11 int minimRecursiv(int arr[], int n) {
12     if(n == 1) return arr[0];
13
14     int min_rest = minimRecursiv(arr, n-1);
15     return (arr[n-1] < min_rest) ? arr[n-1] : min_rest;
16 }
17
18 int main() {
19     int n;
20     printf("Numar elemente: ");
21     scanf("%d", &n);
22
23     int arr[n];
24     printf("Introduceti %d elemente:\n", n);
25     for(int i = 0; i < n; i++) {
26         scanf("%d", &arr[i]);
27     }
28
29     printf("Maxim: %d\n", maximRecursiv(arr, n));
30     printf("Minim: %d\n", minimRecursiv(arr, n));
31
32     return 0;
33 }
```

Output

```
Numar elemente: 3
Introduceti 3 elemente:
2
5
3
Maxim: 5
Minim: 2
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9     printf("Introduceti %d elemente:\n", n);
10    for(int i = 0; i < n; i++) {
11        scanf("%d", &arr[i]);
12    }
13
14    int pare = 0, impare = 0;
15    for(int i = 0; i < n; i++) {
16        if(arr[i] % 2 == 0) pare++;
17        else impare++;
18    }
19
20    printf("Pare: %d\nImpare: %d\n", pare, impare);
21
22    return 0;
23 }
```

Output

```
Numar elemente: 3
Introduceti 3 elemente:
3
7
10
Pare: 1
Impare: 2
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9     printf("Introduceti %d elemente:\n", n);
10    for(int i = 0; i < n; i++) {
11        scanf("%d", &arr[i]);
12    }
13
14    int negative = 0;
15    for(int i = 0; i < n; i++) {
16        if(arr[i] < 0) negative++;
17    }
18
19    printf("Elemente negative: %d\n", negative);
20
21    return 0;
22 }
```

Output

```
Introduceti 5 elemente:
-4
-2
1
5
-8
Elemente negative: 3
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int sursa[n];
9     int destinatie[n];
10
11    printf("Introduceti %d elemente:\n", n);
12    for(int i = 0; i < n; i++) {
13        scanf("%d", &sursa[i]);
14    }
15
16    for(int i = 0; i < n; i++) {
17        destinatie[i] = sursa[i];
18    }
19
20    printf("Tabloul copiat: ");
21    for(int i = 0; i < n; i++) {
22        printf("%d ", destinatie[i]);
23    }
24
25    return 0;
26 }
```

Output

```
Numar elemente: 3
Introduceti 3 elemente:
4
-1
87
Tabloul copiat: 4 -1 87
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente initiale: ");
6     scanf("%d", &n);
7
8     int arr[n+1];
9
10    printf("Introduceti %d elemente:\n", n);
11   for(int i = 0; i < n; i++) {
12       scanf("%d", &arr[i]);
13   }
14
15   int valoare, pozitie;
16   printf("Valoare de inserat: ");
17   scanf("%d", &valoare);
18   printf("Pozitie (0-%d): ", n);
19   scanf("%d", &pozitie);
20   for(int i = n; i > pozitie; i--) {
21       arr[i] = arr[i-1];
22   }
23   arr[pozitie] = valoare;
24
25   printf("Tabloul dupa inserare: ");
26   for(int i = 0; i <= n; i++) {
27       printf("%d ", arr[i]);
28   }
29
30   return 0;
31 }
```

Output

```
Numar elemente initiale: 4
Introduceti 4 elemente:
20
10
4
3
Valoare de inserat: 2
Pozitie (0-4): 1
Tabloul dupa inserare: 20 2 10 4 3
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9
10    printf("Introduceti %d elemente:\n", n);
11    for(int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14
15    int pozitie;
16    printf("Pozitie de sters (0-%d): ", n-1);
17    scanf("%d", &pozitie);
18    for(int i = pozitie; i < n-1; i++) {
19        arr[i] = arr[i+1];
20    }
21
22    printf("Tabloul dupa stergere: ");
23    for(int i = 0; i < n-1; i++) {
24        printf("%d ", arr[i]);
25    }
26
27    return 0;
28 }
```

Output

```
Numar elemente: 3
Introduceti 3 elemente:
13
-25
90
Pozitie de sters (0-2): 2
Tabloul dupa stergere: 13 -25
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9
10    printf("Introduceti %d elemente:\n", n);
11    for(int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14
15    printf("Frecventa elementelor:\n");
16
17    for(int i = 0; i < n; i++) {
18        if(arr[i] == -1) continue;
19
20        int count = 1;
21        for(int j = i+1; j < n; j++) {
22            if(arr[i] == arr[j]) {
23                count++;
24                arr[j] = -1;
25            }
26        }
27        printf("%d apare de %d ori\n", arr[i], count);
28    }
29
30 }
```

Output

```
11
3
11
24
Frecventa elementelor:
11 apare de 2 ori
3 apare de 1 ori
24 apare de 1 ori
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9
10    printf("Introduceti %d elemente:\n", n);
11    for(int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14
15    printf("Elemente unice: ");
16
17    for(int i = 0; i < n; i++) {
18        int esteUnic = 1;
19        for(int j = 0; j < n; j++) {
20            if(i != j && arr[i] == arr[j]) {
21                esteUnic = 0;
22                break;
23            }
24        }
25        if(esteUnic) {
26            printf("%d ", arr[i]);
27        }
28    }
29
30    return 0;
31 }
```

Output

```
Numar elemente: 4
Introduceti 4 elemente:
13
13
2
-4
Elemente unice: 2 -4
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9
10    printf("Introduceti %d elemente:\n", n);
11    for(int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14
15    int duplicate = 0;
16
17    for(int i = 0; i < n; i++) {
18        if(arr[i] == -1) continue;
19
20        int count = 1;
21        for(int j = i+1; j < n; j++) {
22            if(arr[i] == arr[j]) {
23                count++;
24                arr[j] = -1;
25            }
26        }
27        if(count > 1) {
28            duplicate += (count - 1);
29        }
30    }
31
32    printf("Numar total de duplicate: %d\n", duplicate);
33
34    return 0;
35 }
```

Output

```
Introduceti 3 elemente:
25
25
1
Numar total de duplicate: 1

...Program finished with exit code 0
Press ENTER to exit console.
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9
10    printf("Introduceti %d elemente:\n", n);
11    for(int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14
15    for(int i = 0; i < n; i++) {
16        for(int j = i+1; j < n; j++) {
17            if(arr[i] == arr[j]) {
18                for(int k = j; k < n-1; k++) {
19                    arr[k] = arr[k+1];
20                }
21                n--;
22                j--;
23            }
24        }
25    }
26
27    printf("Tablou fara duplicate: ");
28    for(int i = 0; i < n; i++) {
29        printf("%d ", arr[i]);
30    }
31
32    return 0;
33 }
```

Output

```
Numar elemente: 4
Introduceti 4 elemente:
13
-2
24
13
Tablou fara duplicate: 13 -2 24
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n1, n2;
5
6     printf("Dimensiune primul tablou: ");
7     scanf("%d", &n1);
8     int arr1[n1];
9     printf("Introduceti %d elemente:\n", n1);
10    for(int i = 0; i < n1; i++) {
11        scanf("%d", &arr1[i]);
12    }
13
14    printf("Dimensiune al doilea tablou: ");
15    scanf("%d", &n2);
16    int arr2[n2];
17    printf("Introduceti %d elemente:\n", n2);
18    for(int i = 0; i < n2; i++) {
19        scanf("%d", &arr2[i]);
20    }
21
22    int arr3[n1 + n2];
23    for(int i = 0; i < n1; i++) {
24        arr3[i] = arr1[i];
25    }
26    for(int i = 0; i < n2; i++) {
27        arr3[n1 + i] = arr2[i];
28    }
29
30    printf("Tabloul imbunatit: ");
31    for(int i = 0; i < n1 + n2; i++) {
32        printf("%d ", arr3[i]);
33    }
34
35    return 0;
36 }
```

Output

```
13
-2
24
13
Tablou fara duplicate: 13 -2 24
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9
10    printf("Introduceti %d elemente:\n", n);
11   for(int i = 0; i < n; i++) {
12       scanf("%d", &arr[i]);
13   }
14   for(int i = 0; i < n/2; i++) {
15       int temp = arr[i];
16       arr[i] = arr[n-i-1];
17       arr[n-i-1] = temp;
18   }
19
20   printf("Tablou inversat: ");
21   for(int i = 0; i < n; i++) {
22       printf("%d ", arr[i]);
23   }
24
25   return 0;
26 }
```

Output

```
Introduceti 3 elemente:
13
19
2
Tablou inversat: 2 19 13
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9
10    printf("Introduceti %d elemente:\n", n);
11    for(int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14    int pare[n], impare[n];
15    int count_pare = 0, count_impare = 0;
16
17    for(int i = 0; i < n; i++) {
18        if(arr[i] % 2 == 0) {
19            pare[count_pare++] = arr[i];
20        } else {
21            impare[count_impare++] = arr[i];
22        }
23    }
24    printf("Elemente pare: ");
25    for(int i = 0; i < count_pare; i++) {
26        printf("%d ", pare[i]);
27    }
28    printf("\nElemente impare: ");
29    for(int i = 0; i < count_impare; i++) {
30        printf("%d ", impare[i]);
31    }
32
33    return 0;
34 }
```

Output

```
Introduceti 3 elemente:
3
15
10
Elemente pare: 10
Elemente impare: 3 15
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7
8     int arr[n];
9
10    printf("Introduceti %d elemente:\n", n);
11    for(int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14
15    int element;
16    printf("Element de cautat: ");
17    scanf("%d", &element);
18
19    printf("Pozitii gasite: ");
20    int gasit = 0;
21
22    for(int i = 0; i < n; i++) {
23        if(arr[i] == element) {
24            printf("%d ", i);
25            gasit = 1;
26        }
27    }
28
29    if(!gasit) {
30        printf("Elementul nu a fost gasit!");
31    }
32
33    return 0;
34 }
```

Output

```
Introduceti 4 elemente:
14
18
21
28
Element de cautat: 21
Pozitii gasite: 2
```

Input

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Numar elemente: ");
6     scanf("%d", &n);
7     int arr[n];
8     printf("Introduceti %d elemente:\n", n);
9     for(int i = 0; i < n; i++) {
10         scanf("%d", &arr[i]);
11     }
12     int alegere;
13     printf("1 - Crescator\n2 - Descrescator\nAlegere: ");
14     scanf("%d", &alegere);
15     for(int i = 0; i < n-1; i++) {
16         for(int j = 0; j < n-i-1; j++) {
17             if((alegere == 1 && arr[j] > arr[j+1]) ||
18                 (alegere == 2 && arr[j] < arr[j+1])) {
19                 int temp = arr[j];
20                 arr[j] = arr[j+1];
21                 arr[j+1] = temp;
22             }
23         }
24     }
25     printf("Tablou sortat: ");
26     for(int i = 0; i < n; i++) {
27         printf("%d ", arr[i]);
28     }
29
30     return 0;
31 }
```

Output

```
-2
24
37
11
9
1 - Crescator
2 - Descrescator
Alegere: 1
Tablou sortat: -2 9 11 24 37
```

