

Vectori – citire din casete de text

Conținut

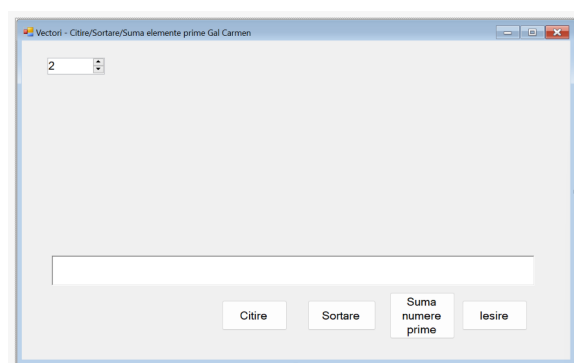
1. Design-ul aplicației:.....1
2. Funcționalitate program.....1
3. Generare casete de text în funcție de valoarea citita pentru n de la butonul NumericUpDown...2
4. Butonul de sortare.....3
5. Butonul pentru determinarea sumei numerelor prime și butonul de ieșire:.....4

Se citește n, număr natural, iar apoi dintr-un vector de casete de text se citesc elementele vectorului, numere întregi. Afișarea rezultatelor se va face într-un RichTextBox. Se cere:

- Sortarea elementelor vectorului și afișarea lor în RichTextBox
- Suma elementelor prime din vector

1. Design-ul aplicației:

- inserarea unui Control NumericUpDown pentru introducerea valorii minime/maxime

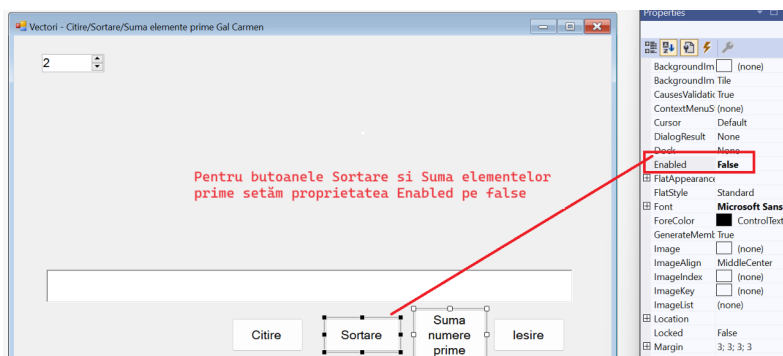


- inserarea butoanelor pentru operațiile de citire vector, sortare, suma elemente prime și ieșirea din aplicație.

-un richTextBox pentru afișarea rezultatelor (inițial inaccesibil)

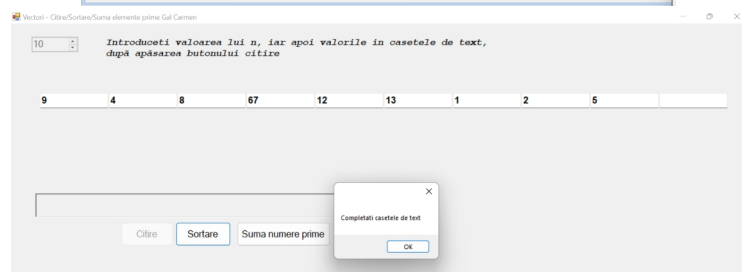
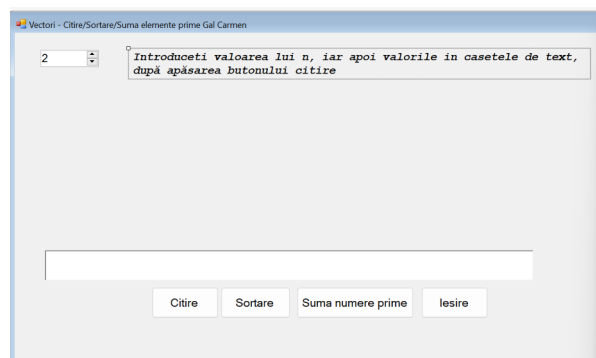
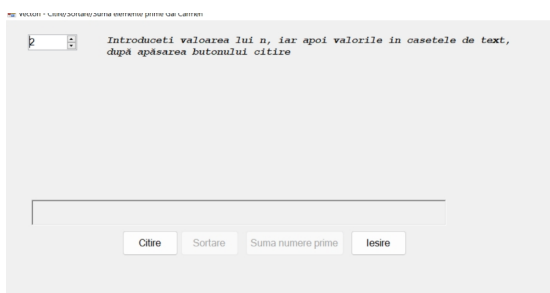
-un label pentru informare

La început butoanele de sortare, suma numere prime, richTextBox sunt inaccesibile

**2. Funcționalitate program**

La început va fi activ butonul numeric UpDown de unde se alege valoarea lui n și butonul de citire și de ieșire:

(proprietatea Enabled pe true sau false, adică accesibil sau inaccesibil).

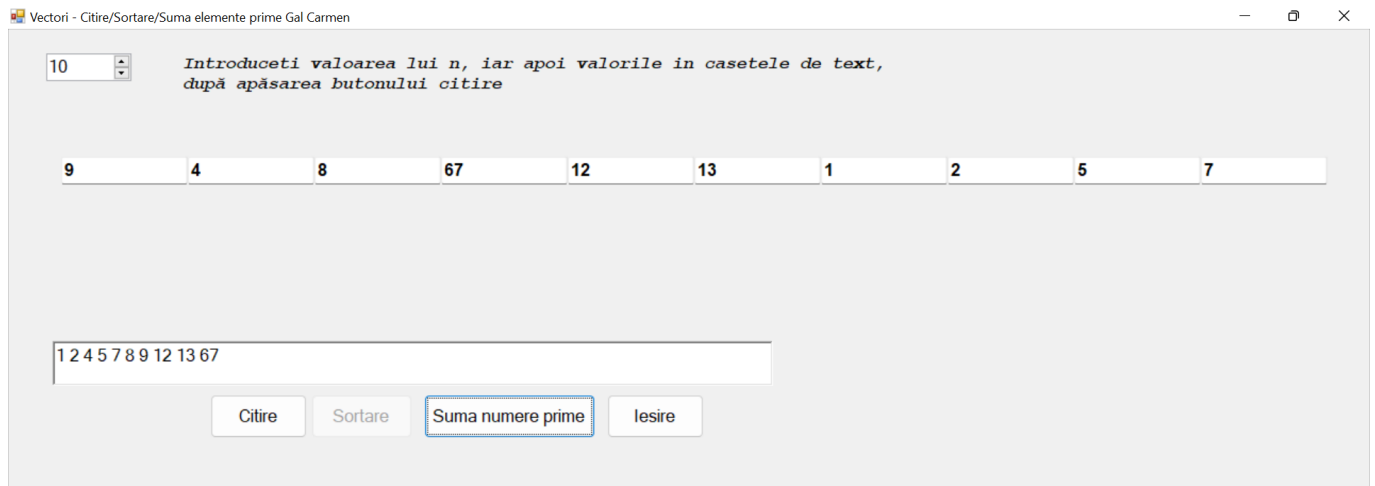


După selectarea

valorii lui n și apăsarea butonului Citire se activează butonul pentru sortare și determinarea sumei. Se introduc valorile în casetele de text. Dacă rămân casete

Prof Carmen Gal

de text necompletate, și se apasă unul din butoanele de sortare sau determinare a sumei apare un mesaj de avertizare. Butonul de citire devine inaccesibil.

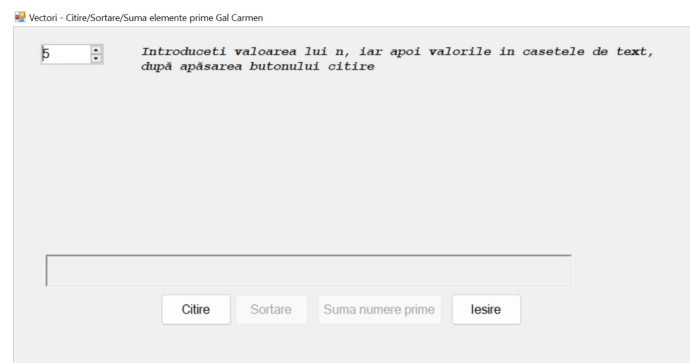
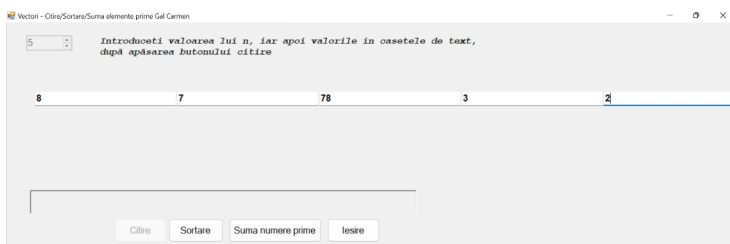


La apăsarea butonului de Sortare și completarea tuturor casetelor de text, valorile sortate vor fi afișate în RichTextBox care devine accesibil. La fel, și controlul NumericUpDown devine accesibil.

La apăsarea butonului Sumă numere prime se va afișa suma numerelor prime din vector.



Dacă se dorește sortarea unui alt vector, prin alegerea unei alte valori pentru n vor fi generate alte casete de text corespunzător noului n



3. Generare casete de text în funcție de valoarea citita pentru n de la butonul NumericUpDown

a) Declarații de variabile și introducerea codului la controlul NumericUpDown:

```
eteText_Gal_Carmen
public partial class Form1 : Form
{
    int n=2,s=0;
    TextBox[] txt=new TextBox[21];
    int[] a=new int[21];
    bool generare = false;
    1 reference
    public Form1()
    {
        InitializeComponent();
    }

    1 reference
    private void numericUpDown1_ValueChanged(object sender, EventArgs e)
    {
        button1.Enabled = true;
        button2.Enabled = false;
        button3.Enabled = false;
        if (generare == true)
        {
            RemoveTextBox(); //daca au fost generate butoanele acestea sunt indepartate
            richTextBox1.Clear(); //curatare richTextBox
            richTextBox1.Enabled = false;
        }
    }
}
```

b) Prima dată introducem codul la butonul de citire:

```
private void button1_Click(object sender, EventArgs e)
{
    //butonul citire pentru generarea casetelor de text in vederea citirii
    if (generare == true)
        RemoveTextBox(); //se apeleaza functia de indepartare a casetelor de text generate anterior
    n = Convert.ToInt32(numericUpDown1.Value); //preluam valoarea lui n
    generare = true;
    AddTextBox(); //apelăm functia pentru generarea casetelor de text
    button2.Enabled = true;
    button3.Enabled = true;
    button1.Enabled = false;
    numericUpDown1.Enabled = false;
}
```

c) Funcțiile pentru adăugarea/îndepărtarea casetelor de text

```
private void RemoveTextBox()
{
    //functia de indepartare a casetelor de text generate anterior
    for (int i = 1; i <= n && generare==true; i++)
        Controls.Remove(txt[i]);
}

private void AddTextBox()
{
    //functia de generare a casetelor de text in functie de n-ul citit
    int w, h;
    if (generare == true)
    {
        h = 100; //latimea si inaltimea casetei de text
        w = this.Width / n - 10; //determinata in functie de latimea si inaltimea formularului
        for (int i = 1; i <= n; i++)
        {
            txt[i] = new TextBox();
            txt[i].Width = w; //latime
            txt[i].Height = h; //inaltime
            txt[i].Top = 120; //alinie fata de marginea de sus
            txt[i].Left = w * (i - 1) + 50; //alinie fata de marginea stanga
            txt[i].Font = new Font(new FontFamily("Arial"), 12, FontStyle.Bold);
            Controls.Add(txt[i]);
        }
    }
}
```

4. Butonul de sortare

```
Form1.cs [Design]
vector1_CitireVectorCaseteText_Gal_Carmen
2 references
bool complete()
{
    //functia care verifică dacă au fost completate toate casetele de text
    int b; //si daca valorile introduse pot fi convertite in numere intregi
    if (generare == true)
        for (int i = 1; i <= n; i++)
            if (txt[i].Text == "" || int.TryParse(txt[i].Text, out b) == false) return false;
    return true;
}

1 reference
private void Sort()
{
    int aux; richTextBox1.Text = "";
    for (int i = 1; i <= n; i++)
        a[i] = Convert.ToInt32(txt[i].Text);
    for (int i = 1; i < n; i++) //sortare
        for (int j = i + 1; j <= n; j++)
            if (a[i] > a[j])
            {
                aux = a[i]; a[i] = a[j]; a[j] = aux;
            }
    for (int i = 1; i <= n; i++) //afisare rezultat in richTextBox1
        richTextBox1.Text += Convert.ToString(a[i]) + " ";
}

1 reference
private void button2_Click(object sender, EventArgs e)
{
    //butonul de sortare
    if (complete() == true) //se verifica daca casetele de text au fost completate
    {
        //si cu valori de tip intreg
        Sort();
        numericUpDown1.Enabled = true;
        button1.Enabled = true;
        button2.Enabled = false;
        richTextBox1.Enabled = true;
    }
    else MessageBox.Show("Completati casetele de text cu numere intregi");
}
```

TryParse(sir, out val_intreg) returnează true dacă valoarea introdusă în caseta de text poate fi convertită într-un număr de tip întreg, altfel false.

Dacă casetele de text au fost completate corect se apelează funcția de sortare Sort unde șirurile de caractere sunt convertite în valori numerice. Se realizează sortarea, iar apoi valorile sortate sunt transformate din nou în șiruri de caractere și depuse în RichTextBox.

5. Butonul pentru determinarea sumei numerelor prime și butonul de ieșire:

```
Form1.cs [Design]
titireVectorCaseteText_Gal_Carmen
vectori1_Ci

1 reference
private void suma()
{
    richTextBox1.Text = "";
    int ok;
    for (int i = 1; i <= n; i++)
        a[i] = Convert.ToInt32(txt[i].Text);
    s = 0;
    for(int i=1;i<=n;i++)
    {
        ok = 1;
        if (a[i] < 2) ok = 0;
        for (int d = 2; d * d <= a[i] && ok==1; d++)
            if (a[i] % d == 0) ok = 0;
        if (ok == 1) s += a[i];
    }
    richTextBox1.Text = Convert.ToString(s);
}

1 reference
private void button4_Click(object sender, EventArgs e)
{
    Application.Exit();
}

1 reference
private void button3_Click(object sender, EventArgs e)
{
    //butonul de determinare suma numere prime
    if (complete() == true)
    {
        suma();
        numericUpDown1.Enabled = true;
        button1.Enabled = true;
        button3.Enabled = false;
        richTextBox1.Enabled = true;
    }
    else MessageBox.Show("Completati casetele de text");
}
}
```