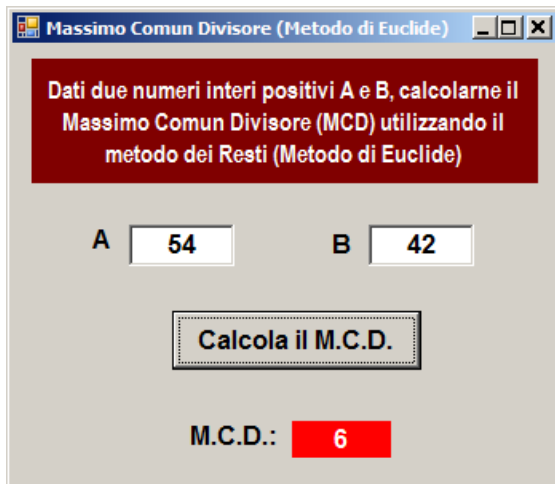


Esempio Pilota:
MassimoComunDivisoreConResti

Definizione della Form



Label txtA

Label txtB

Button plsCalcola

Label lbIMCD

Codice della Form

```
using System;
using System.Windows.Forms;

namespace MassimoComunDivisoreConResti
{
    public partial class frmMCDconResti : Form
    {
        public frmMCDconResti()
        {
            InitializeComponent();
        }

        private void plsCalcola_Click(object sender, EventArgs e)
        {
            // acquisisci A e B dalle caselle di testo ...
            int A = Convert.ToInt16(txtA.Text);
            int B = Convert.ToInt16(txtB.Text);

            // il "Metodo di Euclide" per calcolare l'MCD,
            // ad ogni passo, sostituisce, al numero maggiore, il resto
            // della divisione fra il maggiore e il minore.
            // Prosegue così finchè uno dei due numeri non si azzerava:
            // il MCD è l'altro numero, ossia quello che non si è azzerato.

            // SCELTA DEL TIPO DI ITERAZIONE:
            // Si usa un ciclo while, perchè non è noto a priori il numero di ripetizioni.
            // Inoltre esistono i casi particolari A=0 o B=0 che esigono 0 ripetizioni.

            while ((A != 0) && (B != 0)) // mentre A e B non si sono ancora azzerati
            {
                if (A > B)
                    A = A % B; // ... A è più grande: A diventa il resto di A/B
                else
                    B = B % A; // ... B è più grande: B diventa il resto di B/A
            }

            // CASI PARTICOLARI:
            // Se A=0 o B=0, il ciclo ripete 0 volte: meglio, si evita una divisione per 0!
            // In effetti questo caso non ha senso: ci vorrebbe un if "protettivo" ..

            // Visualizza l'MCD ...

            if (A == 0)
                // A si è azzerato. B è il MCD ...
                lbIMCD.Text = Convert.ToString(B);
            else
                // B si è azzerato. A è il MCD ...
                lbIMCD.Text = Convert.ToString(A);

            // Di seguito trovi la versione OTTIMIZZATA del Metodo di Euclide.
            // Fa in modo che A sia sempre il maggiore. Se, inizialmente, A<B,
            // il primo passo del ciclo ha l'effetto di "scambiare" i valori di A e B (!!!)

            // while (B != 0) ...ripeti mentre B non si è ancora azzerato
            // { R = A % B; ...calcola il resto di A/B
            //   A = B; ...sposta B in A perchè B sarà il maggiore al prossimo ciclo
            //   B = R; } ...spostato il valore di B in A, puoi porre il resto in B
            }
        }
    }
}
```