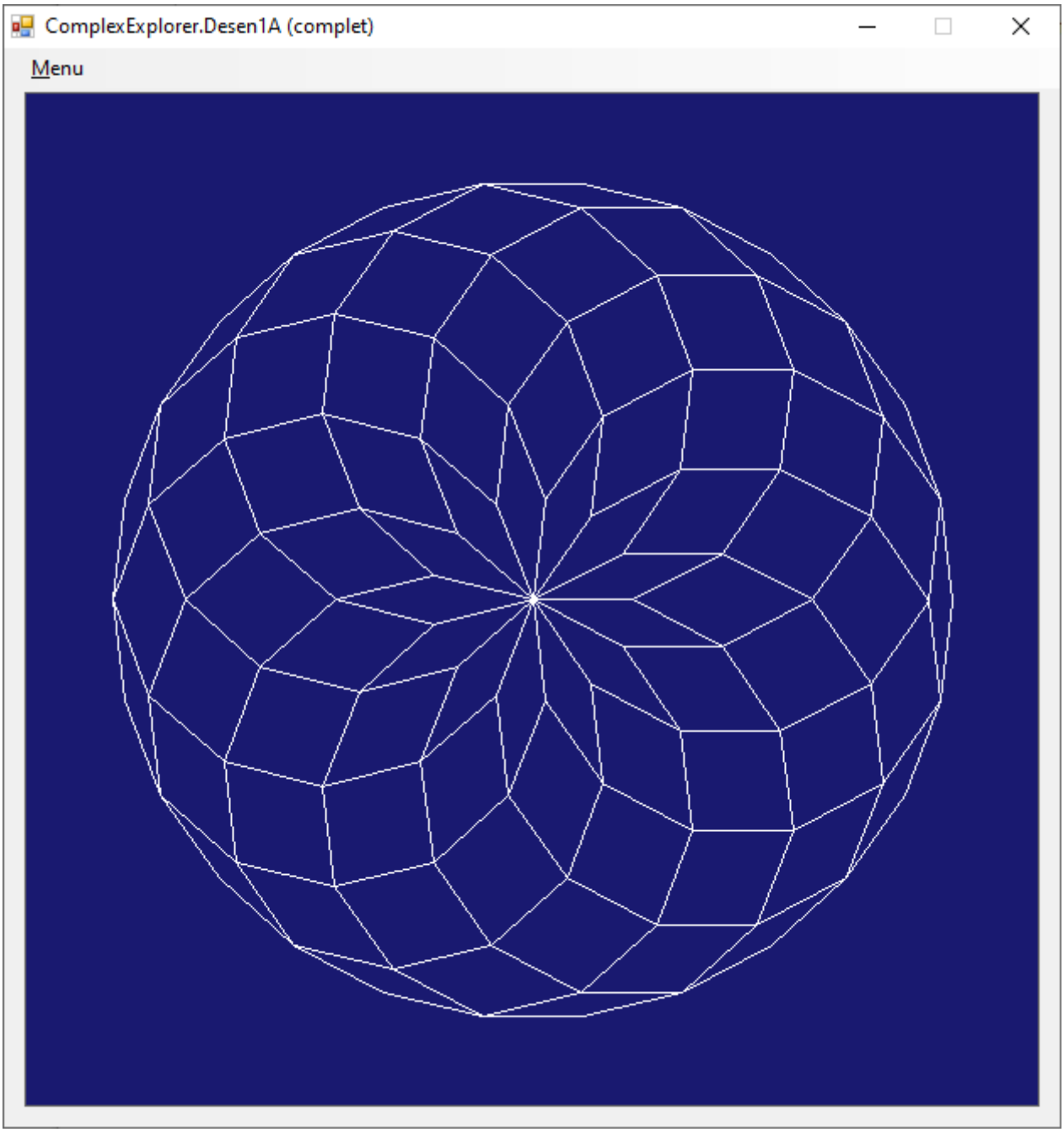
Tablouri în C#

O întrebare: de ce clasele `Desen1A`, `Desen2A` și `Desen1B` de mai jos execută același desen dar nu și clasa `Desen2B`?



```
  public class Desen1A : ComplexForm
  {
      public override void makeImage()
      {
          setXminXmaxYminYmax(-10, 10, -10, 10);
          double R = 2;
          int N = 13;
          Complex[] A = new Complex[2 * N + 1];
          for (int k = 0; k < N; k++)
          {
              A[2 * k] = Complex.setRoTheta(R, 2 * k * Math.PI / N);
              A[2 * k + 1] = 0;
          }
          A[2 * N] = A[0];
          for (int nr = 0; nr <= 6; nr++)
          {
              Complex[] B = new Complex[2 * N + 1];
              for (int k = 0; k < N; k++)
              {
                  setLine(A[2 * k], A[2 * k + 1], PenColor);
                  setLine(A[2 * k + 1], A[2 * k + 2], PenColor);
                  Complex b = A[2 * k] + A[2 * k + 2] - A[2 * k + 1];
                  B[2 * k] = b;
                  B[2 * k + 1] = A[2 * k + 2];
              }
              B[2 * N] = B[0];
              A = B;
              resetScreen();
          }
      }
  }
/*************************************/

  public class Desen2A : ComplexForm
  {
      public override void makeImage()
      {
          setXminXmaxYminYmax(-10, 10, -10, 10);
          double R = 2;
          int N = 13;
          Complex[] A = new Complex[2 * N + 1];
          for (int k = 0; k < N; k++)
          {
              A[2 * k] = Complex.setRoTheta(R, 2 * k * Math.PI / N);
              A[2 * k + 1] = 0;
          }
          A[2 * N] = A[0];
          for (int nr = 0; nr <= 6; nr++)
          {
              Complex[] B = new Complex[2 * N + 1];
              for (int k = 0; k < N; k++)
              {
                  setLine(A[2 * k], A[2 * k + 1], PenColor);
                  setLine(A[2 * k + 1], A[2 * k + 2], PenColor);
                  B[2 * k + 1] = A[2 * k + 2];    //diferit fata de Desen1A
                  Complex b = A[2 * k] + A[2 * k + 2] - A[2 * k + 1];
                  B[2 * k] = b;

              }
              B[2 * N] = B[0];
              A = B;
              resetScreen();
          }
      }
  }
```

```
/************************************/
    public class Desen1B : ComplexForm
    {
        public override void makeImage()
        {
            setXminXmaxYminYmax(-10, 10, -10, 10);
            double R = 2;
            int N = 13;
            Complex[] A = new Complex[2 * N + 1];
            Complex[] B = new Complex[2 * N + 1];   //diferit fata de Desen1A
            for (int k = 0; k < N; k++)
            {
                A[2 * k] = Complex.setRoTheta(R, 2 * k * Math.PI / N);
                A[2 * k + 1] = 0;
            }
            A[2 * N] = A[0];
            for (int nr = 0; nr <= 6; nr++)
            {
                for (int k = 0; k < N; k++)
                {
                    setLine(A[2 * k], A[2 * k + 1], PenColor);
                    setLine(A[2 * k + 1], A[2 * k + 2], PenColor);
                    Complex b = A[2 * k] + A[2 * k + 2] - A[2 * k + 1];
                    B[2 * k] = b;
                    B[2 * k + 1] = A[2 * k + 2];
                }
                B[2 * N] = B[0];
                A = B;
                resetScreen();
            }
        }
    }
    /************************************/
    public class Desen2B : ComplexForm
    {
        public override void makeImage()
        {
            setXminXmaxYminYmax(-10, 10, -10, 10);
            double R = 2;
            int N = 13;
            Complex[] A = new Complex[2 * N + 1];
            Complex[] B = new Complex[2 * N + 1];   //diferit fata de Desen2A
            for (int k = 0; k < N; k++)
            {
                A[2 * k] = Complex.setRoTheta(R, 2 * k * Math.PI / N);
                A[2 * k + 1] = 0;
            }
            A[2 * N] = A[0];
            for (int nr = 0; nr <= 6; nr++)
            {
                for (int k = 0; k < N; k++)
                {
                    setLine(A[2 * k], A[2 * k + 1], PenColor);
                    setLine(A[2 * k + 1], A[2 * k + 2], PenColor);
                    B[2 * k + 1] = A[2 * k + 2];
                    Complex b = A[2 * k] + A[2 * k + 2] - A[2 * k + 1];
                    B[2 * k] = b;

                }
                B[2 * N] = B[0];
                A = B;
                resetScreen();
            }
        }
    }
```

Răspuns: din același motiv pentru care clasele `Desen1C` și `Desen2C` de mai jos nu execută același desen!

```
public class Desen1C : ComplexForm
 {
     public override void makeImage()
     {
         setXminXmaxYminYmax(-10, 10, -10, 10);
         double R = 2;
         int N = 13;
         Complex[] A = new Complex[2 * N + 1];
         for (int k = 0; k < N; k++)
         {
             A[2 * k] = Complex.setRoTheta(R, 2 * k * Math.PI / N);
             A[2 * k + 1] = 0;
         }
         A[2 * N] = A[0];
         for (int nr = 0; nr <= 6; nr++)
         {
             for (int k = 0; k < N; k++)
             {
                 setLine(A[2 * k], A[2 * k + 1], PenColor);
                 setLine(A[2 * k + 1], A[2 * k + 2], PenColor);
                 Complex b = A[2 * k] + A[2 * k + 2] - A[2 * k + 1];
                 A[2 * k] = b;
                 A[2 * k + 1] = A[2 * k + 2];
             }
             A[2 * N] = A[0];
             resetScreen();
         }
     }
 }
 /*************************************/

 public class Desen2C : ComplexForm
 {
     public override void makeImage()
     {
         setXminXmaxYminYmax(-10, 10, -10, 10);
         double R = 2;
         int N = 13;
         Complex[] A = new Complex[2 * N + 1];
         for (int k = 0; k < N; k++)
         {
             A[2 * k] = Complex.setRoTheta(R, 2 * k * Math.PI / N);
             A[2 * k + 1] = 0;
         }
         A[2 * N] = A[0];
         for (int nr = 0; nr <= 6; nr++)
         {
             for (int k = 0; k < N; k++)
             {
                 setLine(A[2 * k], A[2 * k + 1], PenColor);
                 setLine(A[2 * k + 1], A[2 * k + 2], PenColor);
                 A[2 * k + 1] = A[2 * k + 2];
                 Complex b = A[2 * k] + A[2 * k + 2] - A[2 * k + 1];
                 A[2 * k] = b;
             }
             A[2 * N] = A[0];
             resetScreen();
         }
     }
 }
```