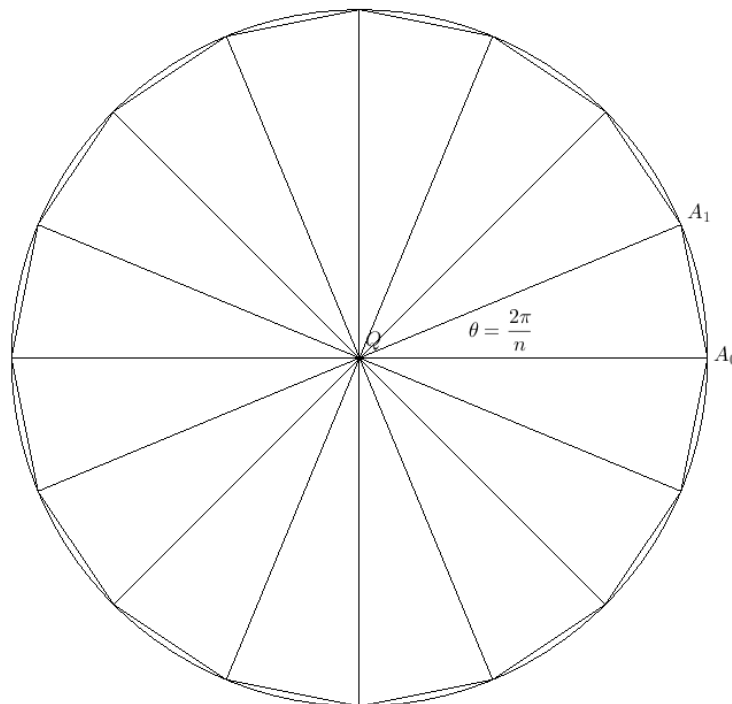


Tema 8

Poligoane regulate

Numim *poligon regulat* un poligon cu toate unghiurile de aceeași măsură și toate laturile de aceeași lungime. Orice poligon regulat cu n laturi se obține prin împărțirea unui cerc în n arce egale. Dacă parcurgem cercul mereu în același sens și unim în ordine cele n puncte obținem un *poligon regulat convex* cu n laturi, numit și n -gon regulat, iar dacă le unim pe sărite, din p în p , cu p prim cu n , obținem un *poligon stelat* cu n laturi, mai precis un n - p -gon regulat.



Fie Q centrul cercului circumscris unui poligon regulat convex cu n laturi și fie A_0 primul vârf al poligonului. Notăm cu litere mici, q și a_0 , afixele lor. Unghiul la centru sub care sunt văzute laturile este $\theta = \frac{2\pi}{n}$. Celelalte vârfuri ale poligonului se obțin rotind pe A_0 în jurul lui Q pe rând cu unghiurile $\theta, 2\theta, 3\theta, \dots$, deci afixele lor sunt

$$a_k = q + \varepsilon^k(a_0 - q), k = 1, 2, \dots, n - 1;$$

unde ε este prima rădăcină de ordin n a unității, $\varepsilon = \cos \theta + i \sin \theta$, și, prin urmare,

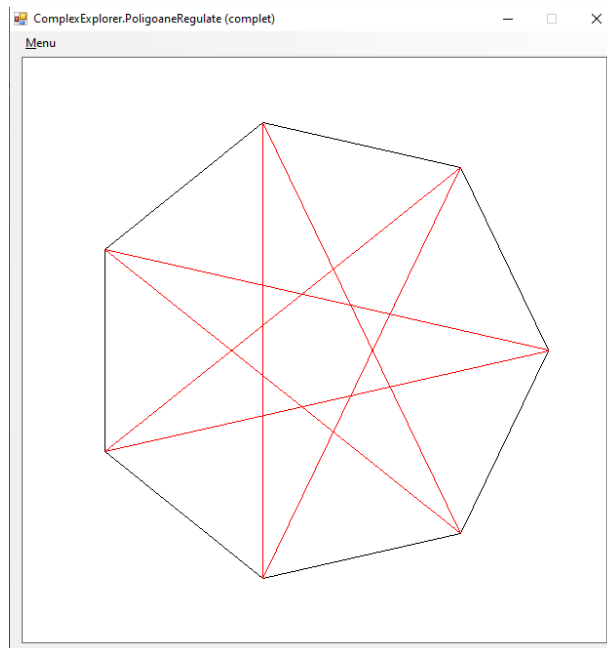
$$\varepsilon^k = \cos k\theta + i \sin k\theta.$$

Poligonul stelat “din p în p ”, n - p -gon-ul regulat, se obține considerând $\theta = p \cdot \frac{2\pi}{n}$.

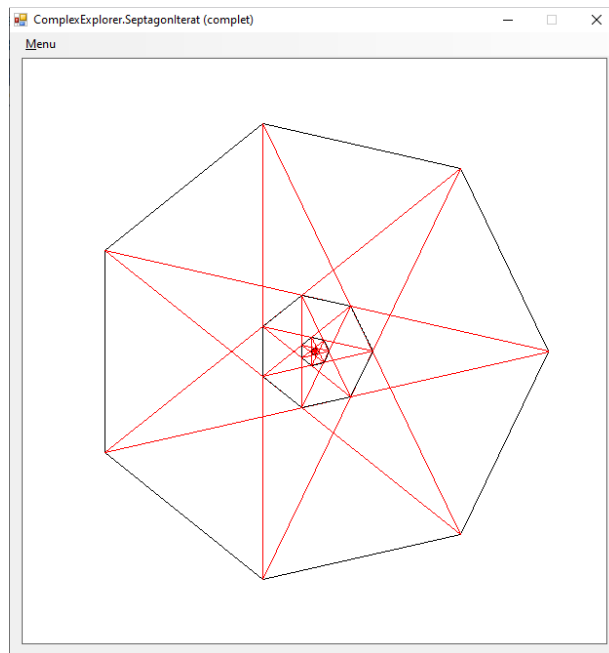
Exercițiul 1. Următorul program trasează cu negru un septagon regulat convex și cu culoarea roșie 7-3-gon-ul stelat corespunzător:

```
public class PoligoaneRegulate : ComplexForm
{
    void traseaza(Complex[] a, Color col)
    {
        for (int k = 1; k < a.Length; k++)
        {
            setLine(a[k - 1], a[k], col);
        }
    }
    Complex[] npGonQA(Complex q, Complex a0, int n, int p = 1)
    { //returneaza varfurile n-p-gonului cu centrul q si primul varf a0
      Complex[] a = new Complex[n + 1];
      double theta = p * 2.0 * Math.PI / n;
      for (int k = 0; k <= n; k++)
      {
          a[k] = q + Complex.setRoTheta(1, k * theta) * (a0 - q);
      }
      return a;
    }
    public override void makeImage()
    {
        setXminXmaxYminYmax(-10, 10, -10, 10);
        ScreenColor = Color.White;

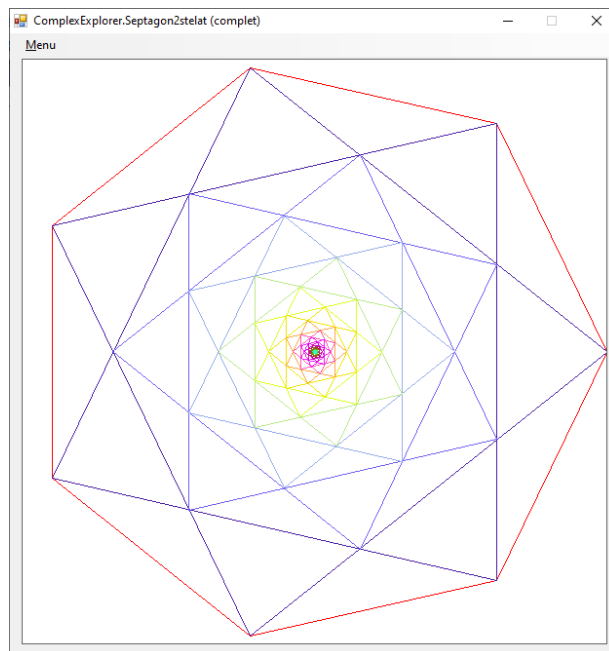
        Complex q = 0;
        Complex a = 8;
        int n = 7;
        traseaza(npGonQA(q, a, n), Color.Black);
        traseaza(npGonQA(q, a, n, 3), Color.Red);
        resetScreen();
    }
}
```



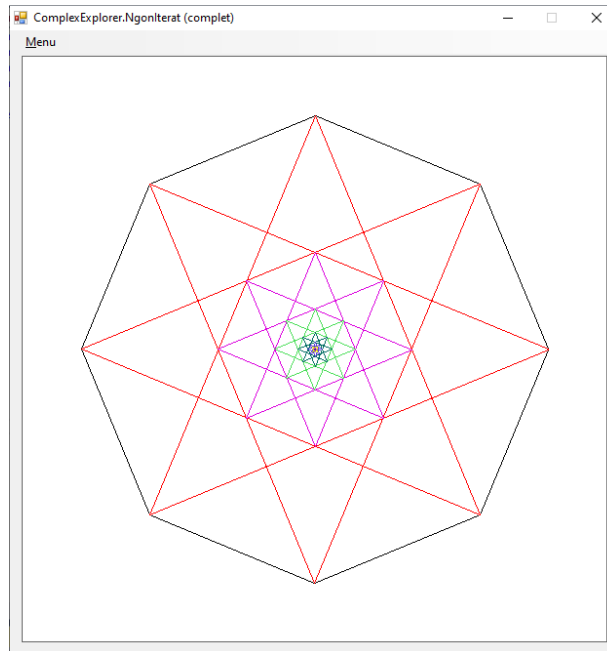
Repetăți această construcție în septagonul convex format în interiorul celui stelat:



Exercițiul 2. Iterați acum un 7-2-gon:



Exercițiul 3. Încercați și cu un octogon regulat:



Exercițiul 4. Programul următor construiește pentagoane regulate pe laturile unui septagon regulat;

```
public class SeptaPentagon : ComplexForm
{
    void traseaza(Complex[] a, Color col)
    {
        for (int k = 1; k < a.Length; k++)
        {
            setLine(a[k - 1], a[k], col);
        }
    }
    Complex[] npGonQA(Complex q, Complex a0, int n, int p = 1)
    { //returneaza varfurile n-p-gonului cu centrul q si primul varf a0
        Complex[] a = new Complex[n + 1];
        double theta = p * 2.0 * Math.PI / n;
        for (int k = 0; k <= n; k++)
        {
            a[k] = q + Complex.setRoTheta(1, k * theta) * (a0 - q);
        }
        return a;
    }
    void bazaApex(Complex zB, Complex zC, double A, out Complex zA,
        bool peStg = true)
    { //calculeaza apexul zA al triunghiului isoscel zB zA zC
        Complex omegaA = Complex.setRoTheta(1, A);
        if (!peStg) omegaA = omegaA.conj;
        zA = (zC - omegaA * zB) / (1 - omegaA);
    }
}
```

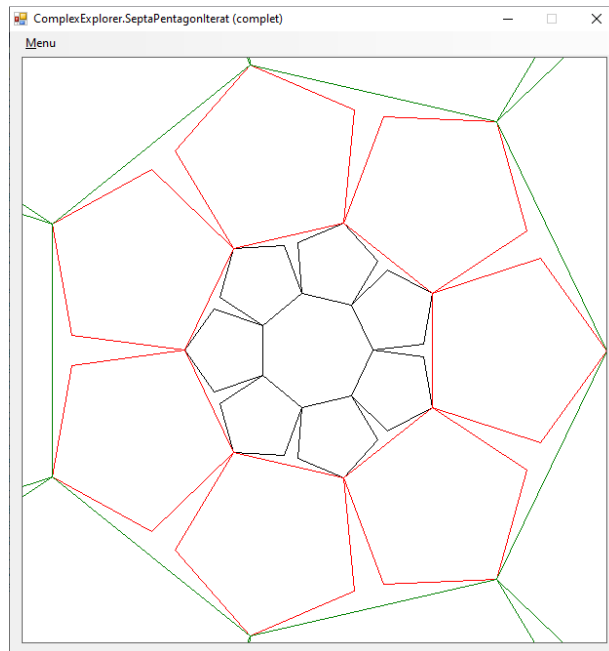
```

public override void makeImage()
{
    setXminXmaxYminYmax(-10, 10, -10, 10);
    ScreenColor = Color.White;

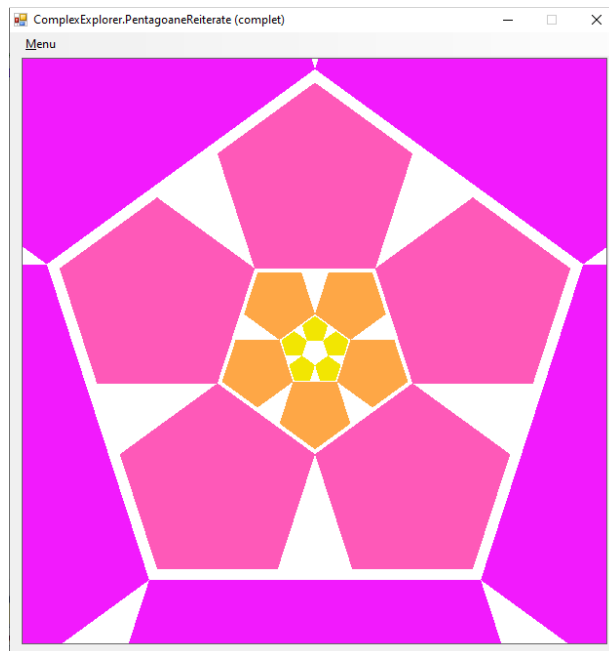
    Complex q = 0;
    Complex a = 2;
    int nBaza = 7;      //septagonul de baza
    int nExt = 5;      //la exterior construim pentagoane
    double thetaExt = 2 * Math.PI / nExt;
    Complex[] baza = npGonQA(q, a, nBaza);
    Complex a0 = baza[0];
    for (int k = 1; k <= nBaza; k++)
    {
        Complex b0 = baza[k];
        Complex q0 = 0; //centrul pentagonului
        bazaApex(a0, b0, thetaExt, out q0, false);
        traseaza(npGonQA(q0, a0, nExt), Color.Black);
        a0 = b0;
    }
    resetScreen();
}
}

```

Completați programul pentru a repeta de câteva ori construcția în exterior:



Exercițiul 5. Desenați pentagoane în jurul unui pentagon:

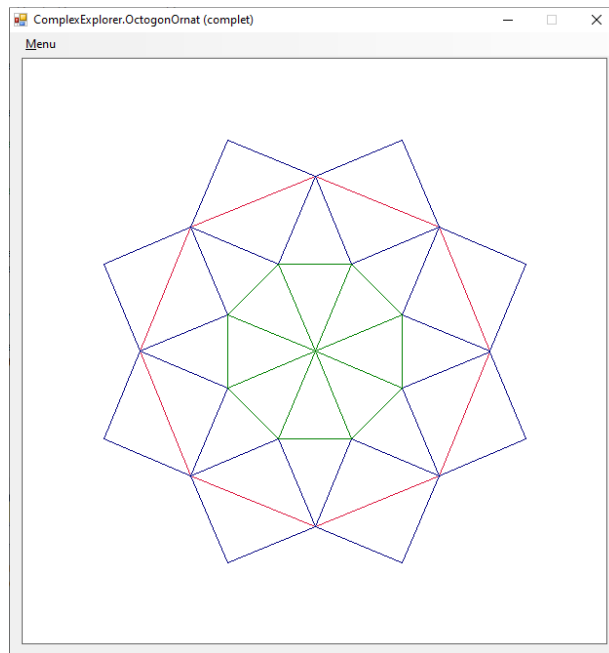


Observație. Pentru umplerea interiorului unei linii poligonale închise fără autointersecții (o curbă Jordan) `Complex[] p` cu o culoare `Color col` se poate folosi perechea de metode

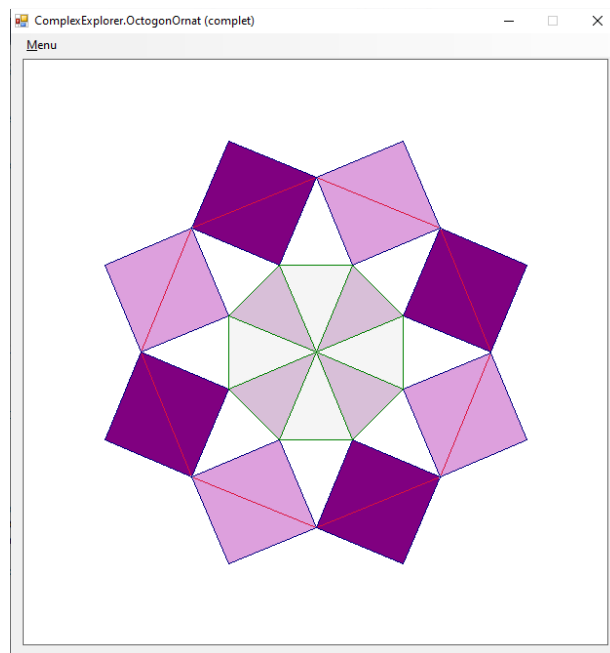
```
public bool esteInInteriorJordan(Complex[] p, Complex z)
{
    //este necesar ca p[N-1]==p[0]
    //poligonul P are N-1 varfuri
    //int N = p.Length;
    double s = 0;
    for (int k = 1; k < p.Length; k++)
    {
        s += ((p[k] - z) / (p[k - 1] - z)).Theta;
    }
    return Math.Abs(s) > 0.5;
}

public void umpleInteriorJordan(Complex[] p, Color col)
{
    //int N = p.Length;
    //este necesar ca p[N-1]==p[0]
    for (int ii = 0; ii <= imax; ii++)
    {
        for (int jj = 0; jj <= jmax; jj++)
        {
            Complex z = getZ(ii, jj);
            if (esteInInteriorJordan(p, z)) setPixel(ii, jj, col);
        }
    }
}
```

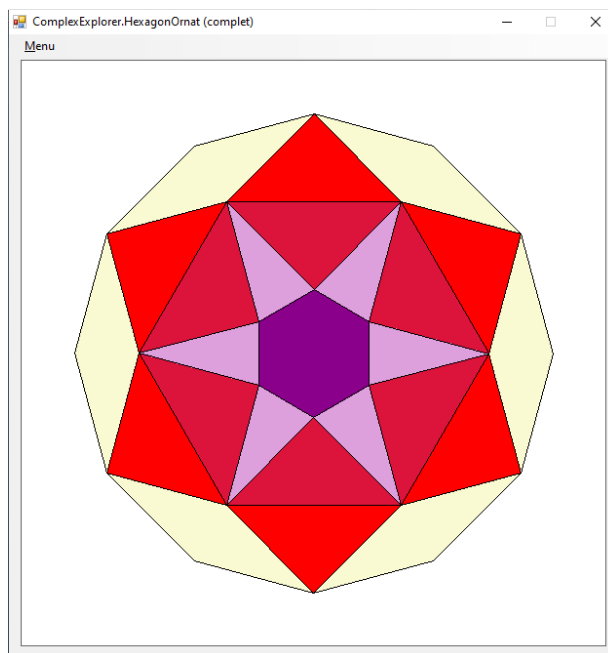
Exercițiul 6. Desenul următor a fost obținut ornând cu pătrate octogonul roșu și cu triunghiuri octogonul verde:



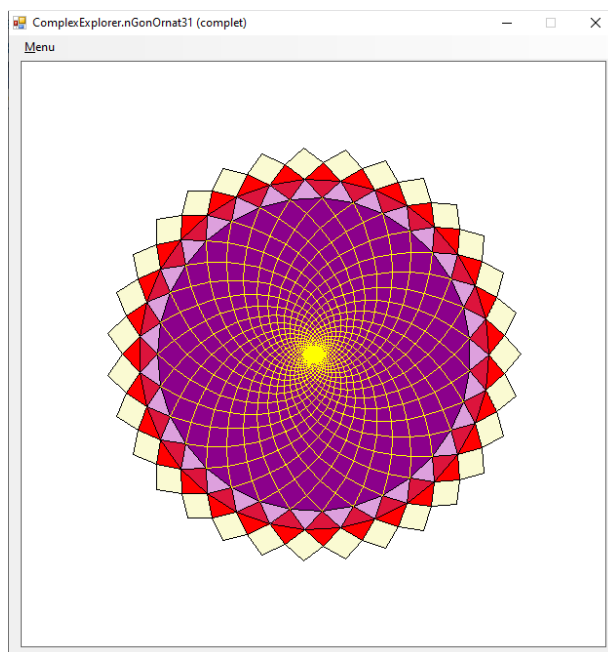
Încercați să-l colorați:



Exercițiul 7. Ornați mai întâi un hexagon regulat



și apoi un 31-gon regulat, pentru a obține o pălărie de floarea-soarelui:



Observație: spiralele interioare au fost obținute cu următoarea metodă, care micșorează și rotește poligonul `Complex[] p` în jurul centrului `Complex q`


```

void roteste(Complex q, Complex[] p, Color col)
{
    Complex omega = Complex.setRoTheta(0.9, 0.1);
    Complex[] pp = (Complex[]) p.Clone();
    for (int k = 0; k < 100; k++)
    {
        for (int j = 0; j < p.Length; j++)
        {
            Complex a = q + omega * (p[j] - q);
            setLine(a, p[j], col);
            p[j] = a;
            Complex aa = q + omega.conj * (pp[j] - q);
            setLine(aa, pp[j], col);
            pp[j] = aa;
        }
    }
}

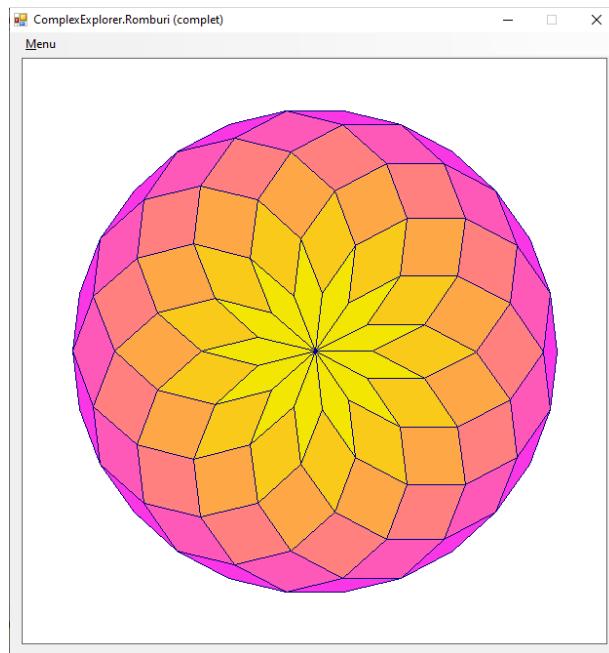
```

Exercițiul 8. Următorul desen a fost obținut trasând toate diagonalele unui n -gon regulat, cu $n = 31$. Încercați și voi:

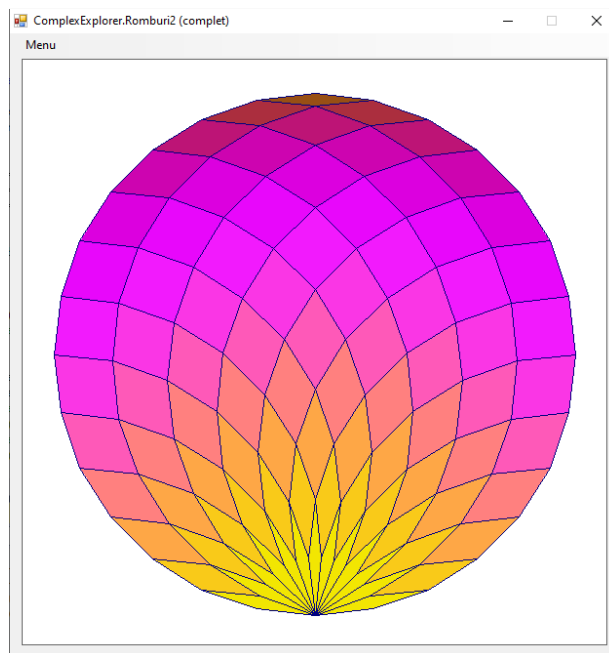


Pentru cazul $n = 32$ vezi: <http://benice-equation.blogspot.ro/2011/10/thirty-two-pointed-star-polygons.html>

Exercițiul 9. a) Următoarea disecție în romburi a unui 26-gon regulat a fost trasată plecând de la cele 13 romburi cu vârful în centru. Încercați și voi:



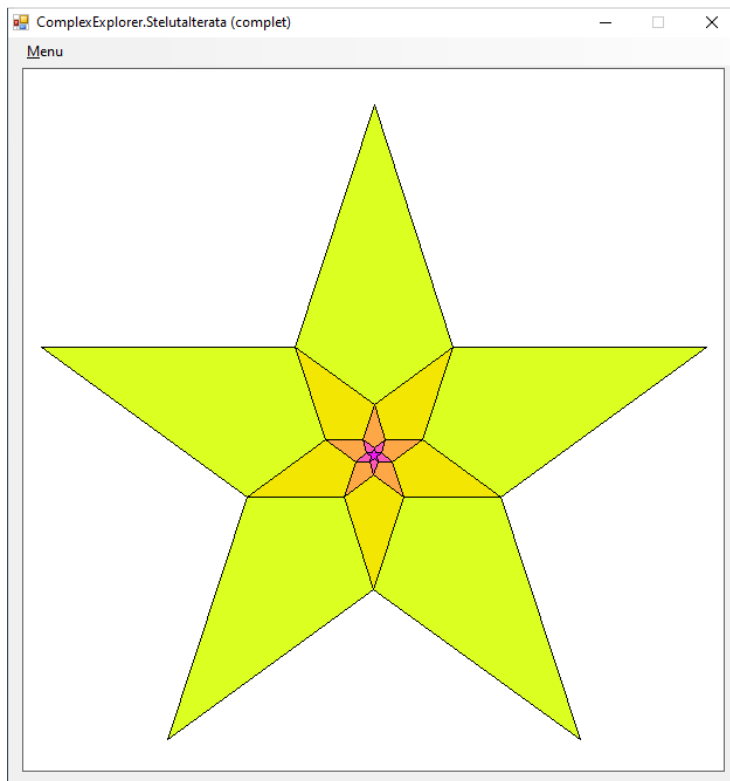
b) Secționați și un 28-gon regulat:



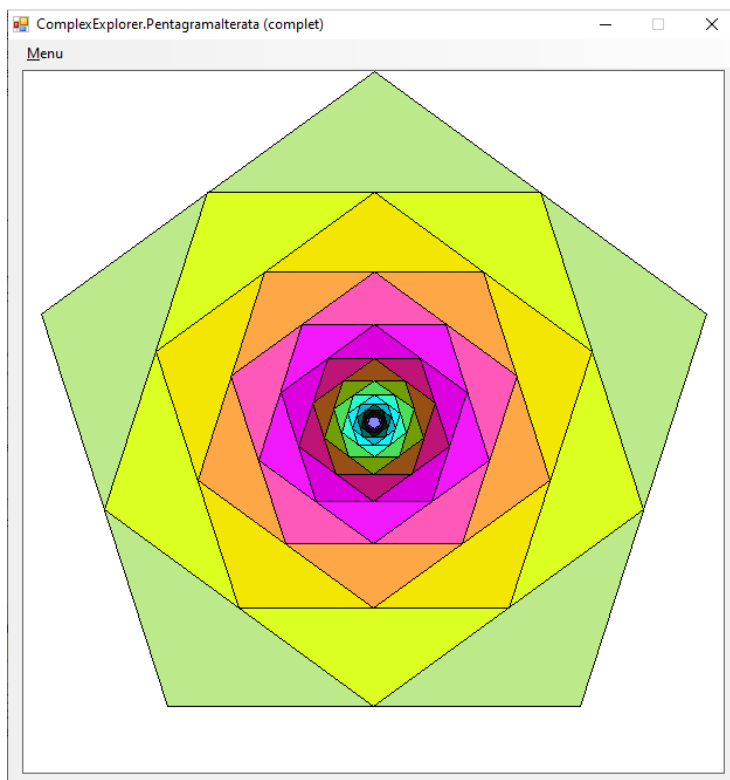
Link: https://en.wikipedia.org/wiki/Regular_polygon

Indicație: Un triunghi abc se completează până la un paralelogram $abcd$ alegând $d = a + c - b$.

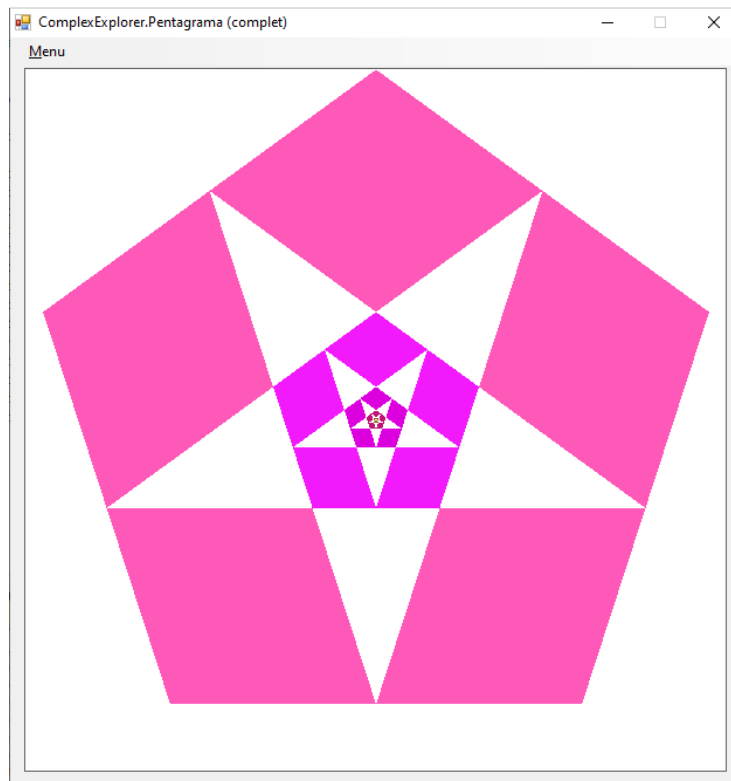
Exercițiul 10. Desenați niște pentagoane regulate:
a)



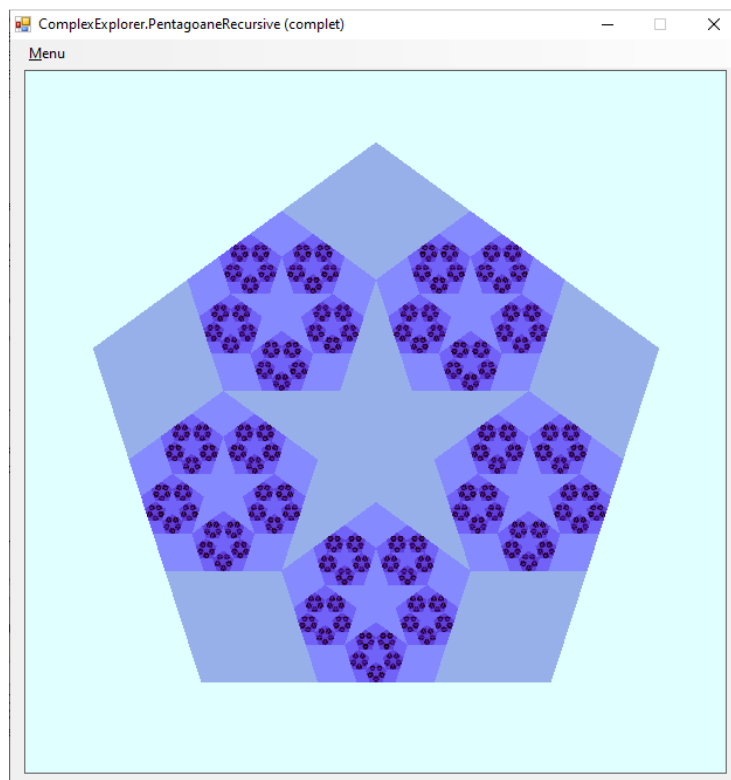
b)



c)



d)



Exercițiul 11. În final, desenați
a) un trandafir



b) și o dalie

