

Aufgaben:

1. KARA soll ein mit Kleeblättern gefülltes Quadrat mit variabler Seitenlänge ablegen. Der Methodenaufruf `quadratZeichnen(5)` soll demnach ein Quadrat mit der Seitenlänge 5 erzeugen.

```
import JavaKaraProgram;

public class Quadrat extends JavaKaraProgram
{ // Anfang von Quadrat

    void turnAround()
    {
        kara.turnLeft();
        kara.turnLeft();
    }

    void legeX(int anzahl)
    {
        for (int i=1; i<=anzahl; i++)
        {
            kara.putLeaf();
            kara.move();
        }
    }

    void geheX(int anzahl)
    {
        for (int i=1; i<=anzahl; i++)
        {
            kara.move();
        }
    }

    void quadratZeichnen (int seitenlaenge)
    { // In der linken oberen Ecke des Quadrates nach rechts starten
        for (int i = 1; i<=seitenlaenge; i++)
        {
            legeX(seitenlaenge);           // Kleeblattzeile ablegen
            turnAround();
            geheX(seitenlaenge);           // und Zeile zuruecklaufen
            kara.turnLeft();               // In naechste Zeile
            kara.move();
            kara.turnLeft();
        }
    }
}
```

```

    }

    public void myProgram()
    { // Anfang von myProgram
      quadratZeichnen (5);
    } // Ende von myProgram
  } // Ende von Quadrat

```

2. KARA soll ein mit Kleeblättern gefülltes Rechteck mit variabler Breite und Höhe ablegen. Der Methodenaufruf `rechteckZeichnen(5,3)` soll demnach ein Rechteck mit der Breite 5 und der Höhe 3 erzeugen.

```

import JavaKaraProgram;

public class Rechteck extends JavaKaraProgram
{ // Anfang von Rechteck

  void turnAround()
  {
    kara.turnLeft();
    kara.turnLeft();
  }

  void legeX(int anzahl)
  {
    for (int i=1; i<=anzahl; i++)
    {
      kara.putLeaf();
      kara.move();
    }
  }

  void geheX(int anzahl)
  {
    for (int i=1; i<=anzahl; i++)
    {
      kara.move();
    }
  }

  void rechteckZeichnen (int breite, int hoehe)
  { // In der linken oberen Ecke des Rechteckes nach rechts starten
    for (int i = 1; i<=hoehe; i++)
    {
      legeX(breite);          // Kleeblattzeile ablegen
      turnAround();
      geheX(breite);          // und Zeile zuruecklaufen
    }
  }
}

```

```

        kara.turnLeft();    // In naechste Zeile
        kara.move();
        kara.turnLeft();
    }
}

public void myProgram()
{ // Anfang von myProgram
    rechteckZeichnen (5,3);
} // Ende von myProgram
} // Ende von Rechteck

```

3. KARA soll ein mit Kleeblättern gefülltes gleichseitiges Dreieck mit variabler Seitenlänge ablegen. Eine Seite des Dreiecks soll waagerecht liegen.

```

import JavaKaraProgram;

public class GSDreieck extends JavaKaraProgram
{ // Anfang von GSDreieck

    void turnAround()
    {
        kara.turnLeft();
        kara.turnLeft();
    }

    void legeX(int anzahl)
    { // Start auf erstem Blatt; Ende hinter letztem Blatt
        for (int i=1; i<=anzahl; i++)
        {
            kara.putLeaf();
            kara.move();
        }
    }

    void geheX(int anzahl)
    {
        for (int i=1; i<=anzahl; i++)
        {
            kara.move();
        }
    }

    void gsDreieckZeichnen (int seite)
    { // In der linken unteren Ecke des gleichseitigen
      // Dreiecks nach rechts starten
      int aktBreite;

```

```

    aktBreite = seite;
    while (aktBreite > 0)
    {
        legeX(aktBreite);        // Kleeblattzeile ablegen
        turnAround();
        geheX(aktBreite-1);      // und Zeile zuruecklaufen
        kara.turnRight();        // In naechste Zeile
        kara.move();
        kara.turnRight();
        aktBreite = aktBreite - 2;
    }
}

public void myProgram()
{ // Anfang von myProgram
    gsDreieckZeichnen (7);
} // Ende von myProgram
} // Ende von GSDreieck

```

4. KARA soll ein mit Kleeblättern umrandetes Rechteck mit variabler Breite und Höhe ablegen.

```

import JavaKaraProgram;

public class RechteckOffen extends JavaKaraProgram
{ // Anfang von RechteckOffen

    void legeX(int anzahl)
    { // Start auf erstem Blatt; Ende hinter letztem Blatt
        for (int i=1; i<=anzahl; i++)
        {
            kara.putLeaf();
            kara.move();
        }
    }

    void rechteckOffenZeichnen (int breite, int hoehe)
    { // In der rechten oberen Ecke des Rechtecks starten
        kara.move();
        for (int i=1; i <= 2; i++)
        {
            legeX(breite-1);    // Kleeblattzeile waagerecht ablegen
            kara.turnRight();
            kara.move();
            kara.turnRight();
            kara.move();
            kara.turnLeft();
        }
    }
}

```

```

        legeX(hoehe-1);      // Kleeblattzeile senkrecht ablegen
        kara.turnRight();
        kara.move();
        kara.turnRight();
        kara.move();
        kara.turnLeft();
    }
}

public void myProgram()
{ // Anfang von myProgram
    rechteckOffenZeichnen (7,5);
} // Ende von myProgram
} // Ende von RechteckOffen

```

5. KARA soll ein mit Kleeblättern gefülltes Quadrat mit variabler Seitenlänge in „Kegelaufstellung“ ablegen. Die Seiten des Quadrates sind Diagonalen in der KARA-Welt. Dazu soll eine Methode `legeKegelreihe(int anzahl)` entwickelt werden, die eine diagonale Kegelreihe ablegt. Eine weitere Methode `zurNaechstenReihe()` ist ebenso zu programmieren.

```

import JavaKaraProgram;

public class KegelQuadrat extends JavaKaraProgram
{ // Anfang von KegelQuadrat

    boolean rechtsUm = true;

    void legeKegelreihe(int anzahl)
    { // Start auf erstem Blatt; Ende hinter letztem Blatt
        for (int i=1; i<=anzahl; i++)
        {
            kara.putLeaf();
            kara.move();
            kara.turnLeft();
            kara.move();
            kara.turnRight();
        }
    }

    void zurNaechstenReihe()
    {
        if (rechtsUm)
        { kara.turnRight();
          kara.move();
          kara.move();
          kara.turnRight();
        }
    }
}

```

```

        }
    else { kara.turnLeft();
           kara.turnLeft();
           kara.move();
           kara.move();
        }
    rechtsUm = !rechtsUm;
}

void kegelLegen (int seite)
{ // In der linken Ecke des Quadrates starten
  for (int i=1; i <= seite; i++)
  {
    legeKegelreihe(seite);
    zurNaechstenReihe();
  }
}

public void myProgram()
{ // Anfang von myProgram
  kegelLegen (3);
} // Ende von myProgram
} // Ende von KegelQuadrat

```