

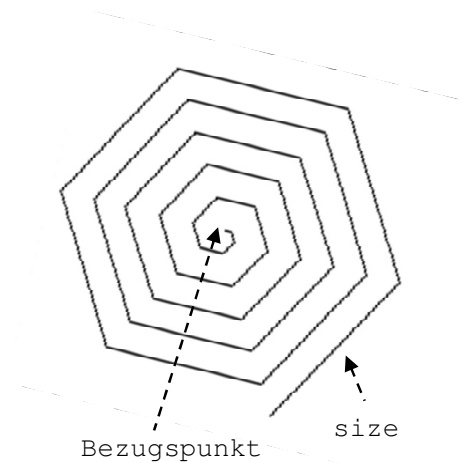
Teil a): Entwurf in Pseudocode

```

private zeichneSpirale()
    stift hoch
    bewege stift bis (xpos | ypos)
    drehe stift bis angle
    stift runter

    wiederhole mit Startwert 3 und
                Inkrement 3 solange
                Schleifenvariable i ≤ size
        bewege stift um i
        drehe stift um -60°

```

**Teil b): Implementierung in Java**

```

private void zeichneSpirale
{
    stift.hoch();
    stift.bewegeBis(xpos, ypos);
    stift.dreheBis(angle);
    stift.runter();

    for(int i = 3; i <= size; i = i + 3)
    {
        stift.bewegeUm(i); // hier wird der Wert der Schleifenvariable i
        stift.dreheUm(-60); // für die aktuelle Linienlänge verwendet
    }
}

```

Teil c): Implementierung in Java / Alternative Schleifenstruktur

Da die erste Variante mit einer *Zählschleife* (*for*) realisiert wurde, wird jetzt in der zweiten Variante eine *Bedingungsschleife* mit Eingangsbedingung (*while*) eingesetzt.

```

private void zeichneSpirale()
{
    stift.hoch();
    stift.bewegeBis(xpos, ypos);
    stift.dreheBis(angle);
    stift.runter();

    int i = 3; // Zählvariable deklarieren und initialisieren

    while(i <= size) // geeignete Endebedingung formulieren
    {
        stift.bewegeUm(i);
        stift.dreheUm(-60);

        i = i + 3; // Inkrement um drei nicht vergessen!
    }
}

```

Teil c): Methode *zeichne()* / *Java-Quellcode*

Da wir jetzt drei Alternativen / Formen haben, reicht eine einfache Verzweigung nicht mehr aus. Eine Lösungsmöglichkeit hierfür ist eine entsprechende ***if/else-Verkettung***.

```
public void zeichne()
{
    if (shape == 1)
        zeichneDreieck();
    else if (shape == 2)
        zeichneSechseck();
    else
        zeichneSpirale();
}
```

Eine solche Verkettung ist aber weder besonders elegant noch übersichtlich, insbesondere, wenn noch mehr Alternativen hinzukommen. Daher gibt es für solche Fälle in *Java* eine Kontrollstruktur für ***Mehrfachverzweigungen***. Eine detaillierte Erläuterung hierzu ist im Kursbereich meiner Seite verfügbar (<http://www.milzners.de/kursbereich/kursbereich.htm#infoSekII>).

Die Anwendung einer Mehrfachverzweigung („*switch/case-Anweisung*“) in der Methode *zeichne()* ist unten dargestellt und weitgehend selbsterklärend.

Beinhaltet die Klassenvariable *shape* einen nicht zulässigen Wert (z.B. per *setShape(...)* gesetzt), wird der *default-Zweig* der Mehrfachverzweigung ausgeführt und auf der *Java-Konsole* eine entsprechende Fehlermeldung ausgegeben.

```
public void zeichne()
{
    switch(shape)
    {
        case 1: zeichneDreieck();
                break;

        case 2: zeichneSechseck();
                break;

        case 3: zeichneSpirale();
                break;

        default: System.out.println("Die shape-Nummer "
                                    + shape
                                    + " ist nicht verfuegbar!\r\n");
    }
}
```