

Aufgabenstellung:

Die LOGO-Funktion ZUFALLSZAHL dient dazu, „Zufallszahlen“ für die Simulation von Zufallsereignissen zu liefern. Mit dieser Funktion soll das Würfeln mit einem gewöhnlichen Spielwürfel simuliert werden.

Dieses „Würfeln“ soll viele Male wiederholt; die Würfelergebnisse sollen statistisch ausgewertet und als Diagramm dargestellt werden.

Durchführung:

1. Jeder einzelne Würfelwurf wird durch die Funktion `wuerfel` simuliert. Die LOGO-Funktion ZUFALLSZAHL :n liefert ganzzahlige „Zufallszahlen“ zwischen 0 und $n-1$. ZZ 6 liefert somit Zahlen zwischen 0 und 5. Durch das Aufaddieren von 1 werden Zahlen zwischen 1 und 6 erzeugt und von `wuerfel` als Rückgabe bereitgestellt:

```
PR wuerfel
  RG 1 + ZZ 6
ENDE
```

2. Die Prozedur `registrierung` ruft jeweils einen Würfelwurf auf und registriert das Ergebnis mit Hilfe von sechs Zählern (pro möglichem Würfelergebnis ein Zähler). Die Zähler sind als globale Variable gestaltet, so dass auch die übergeordneten Prozeduren auf die registrierten Ergebnisse zugreifen können. Dieser „programmiertechnische Makel“ lässt sich erst umgehen, wenn das Sprachkonzept von LOGO auf andere Datentypen erweitert wird.

```
PR registrierung
  SETZE "wurf" wuerfel
  WENN :wurf = 1 DANN SETZE "zaehler1" :zaehler1 + 1
  WENN :wurf = 2 DANN SETZE "zaehler2" :zaehler2 + 1
  WENN :wurf = 3 DANN SETZE "zaehler3" :zaehler3 + 1
  WENN :wurf = 4 DANN SETZE "zaehler4" :zaehler4 + 1
  WENN :wurf = 5 DANN SETZE "zaehler5" :zaehler5 + 1
  WENN :wurf = 6 DANN SETZE "zaehler6" :zaehler6 + 1
ENDE
```

Ergänzend zur Prozedur `registrierung` dient die Prozedur `initialisierung` dazu, die sechs Zähler zu Beginn der Reihe von Würfelversuchen auf 0 zurückzusetzen:

```
PR initialisierung
  LS
  SETZE "zaehler1" 0
  SETZE "zaehler2" 0
  SETZE "zaehler3" 0
  SETZE "zaehler4" 0
  SETZE "zaehler5" 0
  SETZE "zaehler6" 0
ENDE
```

3. Die folgende Prozedur gibt das Ergebnis der mehrfachen Würfelversuche als Klartext im Textfenster von WINLOGO aus:

```
PR textausgabe :anzahl
  DR "Es wurde " DR :anzahl DZ " Mal gewürfelt."
  DR "Die Eins wurde " DR :zaehler1 DZ " Mal gewürfelt."
  DR "Die Zwei wurde " DR :zaehler2 DZ " Mal gewürfelt."
  DR "Die Drei wurde " DR :zaehler3 DZ " Mal gewürfelt."
  DR "Die Vier wurde " DR :zaehler4 DZ " Mal gewürfelt."
  DR "Die Fünf wurde " DR :zaehler5 DZ " Mal gewürfelt."
  DR "Die Sechs wurde " DR :zaehler6 DZ " Mal gewürfelt."
  DZ ""
ENDE
```

4. Die letzte Unterprozedur `diagramm` zeichnet ein einfaches Säulendiagramm (Histogramm) zu den sechs Zählerständen:

```

PR diagramm :anzahl
  SETZE "streckfaktor" ( 200 * 6 ) / :anzahl
  BILD
  SH AUFXY 40 40 RE 90 SA VW 300 RW 300
  VW 30 LI 90 VW :zaehler1 * :streckfaktor RE 90 VW 20 RE 90 VW :zaehler1 * :streckfaktor
  LI 45 RW 4 FÜLLE 0 VW 4 LI 45
  VW 30 LI 90 VW :zaehler2 * :streckfaktor RE 90 VW 20 RE 90 VW :zaehler2 * :streckfaktor
  LI 45 RW 4 FÜLLE 0 VW 4 LI 45
  VW 30 LI 90 VW :zaehler3 * :streckfaktor RE 90 VW 20 RE 90 VW :zaehler3 * :streckfaktor
  LI 45 RW 4 FÜLLE 0 VW 4 LI 45
  VW 30 LI 90 VW :zaehler4 * :streckfaktor RE 90 VW 20 RE 90 VW :zaehler4 * :streckfaktor
  LI 45 RW 4 FÜLLE 0 VW 4 LI 45
  VW 30 LI 90 VW :zaehler5 * :streckfaktor RE 90 VW 20 RE 90 VW :zaehler5 * :streckfaktor
  LI 45 RW 4 FÜLLE 0 VW 4 LI 45
  VW 30 LI 90 VW :zaehler6 * :streckfaktor RE 90 VW 20 RE 90 VW :zaehler6 * :streckfaktor
  LI 45 RW 4 FÜLLE 0 VW 4 LI 45
  VW 30
ENDE

```

Diese Prozedur weist zwei Besonderheiten auf:

- (a) Die Variable `streckfaktor` wird so ausgerechnet, dass das Diagramm, unabhängig von der gewählten Gesamtzahl von Würfelversuchen immer eine Höhe von etwa 200 Bildschirmpunkten hat: Auf jede Zahl entfällt etwa ein Sechstel aller Ergebnisse, daher ist der Streckfaktor der Reziprokwert der Gesamtzahl aller Versuche, geteilt durch $6 \cdot 200$.
- (b) Jede der Doppelzeilen 5 bis 16 des Listings erzeugt eine Säule des Diagramms, deren Höhe in Abhängigkeit vom zugehörigen Zählerinhalt bestimmt wird.

Um diese Säulen schwarz zu füllen, wird der Igel, wenn er in der rechten unteren Ecke der Säule angelangt ist, diagonal rückwärts vier Schritte in das Säuleninnere bewegt, um dort den Füllvorgang mit `FÜLLE` zu bewirken. Anschließend wird der Igel wieder vor auf den Rand bewegt.

5. Die nachfolgende Hauptprozedur ruft die verschiedenen Unterprozeduren nacheinander auf. Mit Hilfe des Übergabeparameters `anzahl` können die Nutzenden dieses Programms die Anzahl der durchzuführenden „Würfelwürfe“ wählen.

`STARTEZUFALL` initialisiert die Startroutine für die Funktion `ZUFALSSZAHL`, damit die „Zufallszahlen“ nicht immer in der gleichen Reihenfolge erzeugt werden:

```

PR wuerfeltest_m_d :anzahl
  initialisierung
  STARTEZUFALL
  WH :anzahl [ registrierung ]
  textausgabe :anzahl
  diagramm :anzahl
ENDE

```

Auswertung und Ausblick:

Das Programm erfüllt die Mindestanforderungen der Aufgabenstellung. Auf die Verwendung globaler Variablen, die gegen ein streng lokales Programmierkonzept verstoßen, wurde bereits hingewiesen.

Folgende Punkte könnten noch verbessert oder ergänzt werden:

- Auf jeden Fall sollte die x-Achse des Säulendiagramms noch beschriftet werden; in einem weiteren Schritt könnte auch die y-Achse noch skaliert werden (allerdings muss diese Skalierung auch flexibel gemäß dem Streckfaktor programmiert werden).
- Das Programm funktioniert einwandfrei nur, wenn mindestens 13 Würfeldurchläufe gefordert werden — sonst wird der gesamte Bildschirm geschwärzt. Dies liegt vermutlich daran, dass bei diesen geringen Versuchszahlen einzelne Säulen so klein bleiben, dass das „Rangieren“ des Igels vor dem Füllen diesen schon wieder aus dem Säuleninneren herausbewegt. Dadurch wirkt der Füllvorgang auf den gesamten Bildschirm. Hier könnte noch Abhilfe geschaffen werden.