



Liebe Schülerin, lieber Schüler,

wir freuen uns, Sie heute zum Landeswettbewerb des Sächsischen Informatikwettbewerbs 2015/2016 begrüßen zu können und wünschen Ihnen viel Erfolg, aber auch Freude bei der Lösung der Aufgaben.

Hier noch einige Hinweise:

### Arbeitszeit

- Für die Lösung der Aufgaben haben Sie 4 Stunden (240 min) Zeit.

### Hilfsmittel

- Als Hilfsmittel sind Standardsoftware, Programmierumgebungen sowie Taschenrechner und Tafelwerk zugelassen.
- **Standardsoftware:**
  - Office-Pakete (Microsoft Office; OpenOffice)
  - Zeichenprogramme (Paint; Gimp; Inkscape)
- **Programmiersprachen (Entwicklungsumgebung):**
  - Pascal (Freepascal; Lazarus)
  - C, C++ (Dev C++)
  - Visual C (Microsoft Visual Studio; NetBeans)
  - Visual C++ (Microsoft Visual Studio; NetBeans)
  - Visual C# (Microsoft Visual Studio)
  - Visual Basic (Microsoft Visual Studio)
  - Python (Eclipse)
  - PHP (Eclipse; NetBeans; xampp [PHP])
  - Java (Eclipse; Java-Editor; NetBeans)
  - Java-Script (HTML Editor Phase 5)
  - Karol (Karol)
  - Scratch (Scratch)

### Bewertung/Organisation

- Für die Aufgabe 1 gibt es 15 Punkte, für die Aufgabe 2 werden 30 Punkte vergeben. **Beachten Sie bitte auch die Punktverteilung auf den Aufgabenzetteln.**
- Es ist wichtig, dass der Lösungsweg deutlich wird.
- **Zu jeder Aufgabe ist ein Teil der Aufgaben auf dem Papier (siehe vorbereitete Zettel) zu lösen.** Beachten Sie bitte, dass auf jedem Zettel nur Lösungen zu der jeweiligen Aufgabe 1 **oder** zur Aufgabe 2 stehen dürfen. Weitere Lösungszettel erhalten Sie bei Bedarf von der Aufsicht. Wenn Sie die Lösung der Aufgaben in eine Datei schreiben, vermerken Sie bitte den Dateinamen auf dem Papier.
- Speichern Sie alle Dateien in den Ordnern Z:\Aufgabe1 bzw. Z:\Aufgabe2. Nur diese Dateien werden bewertet. Sollten Sie ein anderes Arbeitsverzeichnis benutzen, müssen Sie nach Fertigstellung Ihrer Arbeit die Dateien (auch \*.exe) dorthin kopieren.

**Wir wünschen Ihnen viel Erfolg!**



### Zufallszahlen

Scheinbar zufällige Zahlen werden bei einer großen Anzahl von Computerlösungen benötigt. Programmiersprachen und andere Softwarewerkzeuge bieten gewöhnlich Standardfunktionen für solche Umsetzungen. Die Algorithmen, die zur Berechnung solcher zum Einsatz kommen, liefern qualitativ unterschiedliche Ergebnisse.

Mit der folgenden rekursiven Bildungsvorschrift wird eine Folge von natürlichen Zufallszahlen aus einem Intervall  $[0, m-1]$  erzeugt.

$$x_0 = c; c \in \mathbb{N}, c < m$$

$$x_{n+1} = (a \cdot x_n + b) \bmod m$$

Mit mod (Modulo) wird der Rest bei der Division bestimmt, in vielen Programmierumgebungen wird die entsprechende Funktion durch % oder mod ausgedrückt.

#### Aufgaben:

- a) Bestimmen Sie die ersten 4 Glieder der Zahlenfolge für  $m=4$ ,  $b=1$ ,  $a=5$  und  $x_0=2$ . 1 Punkt
- b) Entwickeln Sie eine Computerlösung, die nach Eingabe von  $m$ ,  $b$ ,  $a$ ,  $x_0$  die ersten 1000 Glieder der Zahlenfolge ausgibt. 2 Punkte

In Teilaufgabe b) ist zu erkennen, dass sich die Glieder der Folge nach einer bestimmten Länge wiederholen. Die Maximallänge dieser Perioden ist  $m$ . Nur dann ist gewährleistet, dass jede Zahl aus dem Intervall  $[0, m-1]$  in der Folge genau einmal vorkommt. Nach dem „Satz von Knuth“ wird diese maximale Periodenlänge genau dann erreicht, wenn:

- $b$  und  $m$  teilerfremd sind,
- jeder Primfaktor von  $m$  auch  $a-1$  teilt und
- wenn  $m$  durch 4 teilbar ist, dann auch  $a-1$ .

- c) Bestimmen Sie Werte  $m$ ,  $a$ ,  $b$  und  $x_0$  so, dass  $m > 1000$  ist und sich eine maximale Periodenlänge ergibt. 2 Punkte

Bei typischen Anwendungen zufälliger Zahlenfolgen werden oft Werte aus deutlich kleineren Zahlenbereichen benötigt. Bei Würfelexperimenten zum Beispiel sucht man nach Augenzahlen aus dem Intervall  $[1, 6]$ .

- d) Begründen Sie, warum es für solche Experimente nicht möglich ist, Zufallszahlen mit der Wahl eines kleinen  $m$  (z.B. bei Würfelexperimenten  $m=6$ ) zu erzeugen. Geben Sie eine Möglichkeit an, um bei Wahl eines großen  $m$ , zufällige Zahlen von  $v$  bis  $w$  ( $v < w < m$ ) zu bestimmen. 2 Punkte
- e) Nennen Sie eine Schwäche dieses Verfahrens gegenüber echten Zufallszahlen, die durch Zufallsexperimente gewonnen werden. Wie kann man diese Schwäche durch eine Erweiterung des Algorithmus beseitigen. 1 Punkt
- f) Erweitern Sie Ihre Computerlösung aus Teilaufgabe b) mit Ihren in den Teilaufgaben d) und e) begründeten Veränderungsvorschlägen. Eingabegrößen für diese Computerlösung sollen sein:  $m$ ,  $b$ ,  $a$ ,  $x_0$ ,  $v$  und  $w$ . 3 Punkte

# Gymnasium

## Landeswettbewerb 2015/2016

Klassenstufen 11/12

### 1. Aufgabe

---



Da die Güte der eingesetzten Zufallszahlen, je nach Anwendungsgebiet von entscheidender Bedeutung ist, existieren verschiedene Methoden, wie Folgen solcher Zahlen geprüft werden können. Ein Verfahren hierfür ist der Maximumtest. Dabei werden die Glieder der Folge in Gruppen dreier hintereinander folgender Zahlen  $abc$  zerlegt und es wird geprüft, ob gilt  $b > a$  und  $b > c$ .

- g) Der Erwartungswert der relativen Häufigkeit bei gleich verteilten Zufallszahlen aus  $[0,9]$  im Maximumtest beträgt  $0,285$ . Weisen Sie mit einer vollständigen Darstellung aller Möglichkeiten und der günstigen Möglichkeiten diesen Erwartungswert nach.

2 Punkte

- h) Erstellen Sie eine Computerlösung, die zwei mit unterschiedlichen Verfahrenen erzeugten Folge von 5000 Zufallszahlen aus  $[0, 9]$  durch einen Maximumtest überprüft. Formulieren Sie eine Aussage zur Güte der Zahlenfolgen.

2 Punkte

### Rösselsprung

In einem Rätselbuch wird die in Abbildung 1 gezeigte Aufgabe gestellt.

Beim Rösselsprung (Springer beim Schachspiel) bewegt sich die Figur für einen Zug um zwei Felder in eine beliebige Richtung (links, unten, rechts, oben) und um ein Feld senkrecht dazu. Der nächste Buchstabe nach dem Start oben links (siehe Aufgabe in Abbildung 1) könnte also ein E, ein I oder ein A sein. Ein einmal benutztes Feld darf nicht wieder verwendet werden.

#### 10.1 Osterei mit Rösselsprung

In das Osterei (s. Bild 1) wurden in der Gangart eines Springers beim Schachspiel, d. h. im Rösselsprung, vier eng mit dem Osterfest zusammenhängende Begriffe eingetragen, wobei der Rösselsprung im Feld oben links mit dem Buchstaben E zu beginnen ist. Wie lauten diese vier Begriffe?



Abbildung 1 Quelle: Mildner, Roland: Knifflige Zeiten: Rabenstückverlag 2011

#### Aufgaben:

- a) Geben Sie drei vierbuchstabile Zeichenketten an, die sich in der Abbildung 1 ergeben, wenn oben links gestartet wird. 1 Punkt

- b) Eine etwas einfachere Aufgabe zeigt Abbildung 2. Hier versteckt sich genau ein sinnvolles Wort, wenn jedes Feld genau einmal angesprungen wird. Der Start ist wieder oben links. Nennen Sie dieses Wort und beschreiben Sie Ihr Vorgehen beim Suchen. 5 Punkte

	I	O	
F	R	I	M
T	R	N	E
	K	A	

Abbildung 2

- c) Von einer beliebigen Position aus sind maximal 8 verschiedene Springerzüge möglich. Entwerfen Sie eine Datenstruktur für das Spielfeld und beschreiben Sie einen Algorithmus, der für diese 8 Züge die jeweilige neue Position berechnet. 4 Punkte
- d) Nicht immer sind alle 8 Züge möglich. Beschreiben Sie zwei verschiedene Varianten, mit denen in einem Programm geprüft werden kann, ob ein Zug möglich ist. 4 Punkte
- e) Entwickeln Sie eine Computerlösung, die alle vollständigen Zeichenketten (in denen also jeder der 12 Buchstaben genau einmal vorkommt) in Abbildung 2 findet. Zum Test können Sie die Datei Text2.txt verwenden. Geben Sie alle Lösungen an. 8 Punkte
- f) Für die Lösung der ursprünglichen Aufgabe aus Abbildung 1 muss das Spielfeld im Rösselsprung einmal komplett durchlaufen werden, wobei jedes Feld genau einmal „angesprungen“ wird. Dabei entsteht eine Zeichenkette mit 44 Zeichen. Es gibt sehr viele Lösungen für diese Aufgabe, eine davon ergibt nacheinander gelesen vier Begriffe, die mit Ostern zu tun haben. Prüfen Sie, ob Ihre Computerlösung auch für diese Aufgabe geeignet ist und beschreiben Sie gegebenenfalls mögliche Erweiterungen, mit denen die Aufgabe gelöst werden kann. 4 Punkte
- g) Setzen Sie Ihre Ideen aus Aufgabe f) in eine Computerlösung um. Zum Test können Sie die Datei Text1.txt verwenden. 4 Punkte