

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)
 - Kara (Kara)
 - 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!

Gymnasium

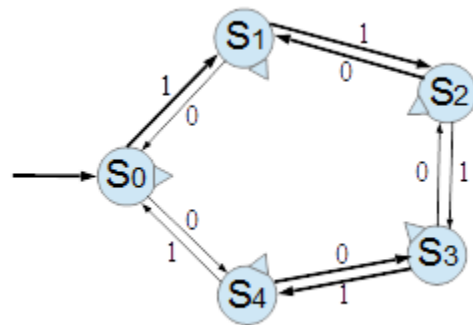
Landeswettbewerb 2015/2016



Wanderball

Am Ende einer Sportstunde ist noch etwas Zeit übrig und der Sportlehrer lässt sich noch schnell ein kleines Ballspiel einfallen: Je 5 Schüler stellen sich im Kreis auf. Alle merken sich eine Richtung rechts/links, in die sie den Ball ihrem Nachbarn zuwerfen wollen. Der Sportlehrer wirft den Ball einem Spieler zu und das Spiel beginnt.

Die Abbildung stellt die Spieler S_0 bis S_4 dar. S_0 bekommt den Ball vom Sportlehrer zugeworfen. Eine „1“ bedeutet, der Spieler gibt den Ball nach links ab, eine „0“ steht für die Abgabe nach rechts.



Beispiel: Spieler S_0 merkt sich die Richtung „links“, S_1 merkt sich „links“, S_2 „rechts“, S_3 „links“ und S_4 „rechts“. Dann lassen sich die gemerkten Richtungen durch ein „Wort“ der Form $\text{richtung}=\{1,1,0,1,0\}$ (Eingabewort) beschreiben. Den Lauf des Balles von Spieler zu Spieler kann ein Ausgabewort z.B. $\text{spiel}=\{1,1,0\}$ wiedergeben. In diesem Beispiel würden sich die Spieler den Ball in der Reihenfolge $S_0 \rightarrow S_1 \rightarrow S_2 \rightarrow S_1$ zuspielen.

Aufgaben:

- Gegeben ist das Eingabewort $\text{richtung_a}=\{0,0,0,0,0\}$. Beschreiben Sie den Spielverlauf und geben Sie die Reihenfolge der Spieler mit Ballbesitz an. Nennen Sie das Ausgabewort $\text{spiel_a}=\{\dots\}$ für ein Spiel, bei dem jeder Spieler den Ball mindestens einmal besitzt und weitergibt. 3 Punkte
- Es existiert ein weiteres Eingabewort $\text{richtung_b}=\{0,\dots\}$, welches ebenfalls jeden Spieler mindestens einmal in Ballbesitz kommen lässt. Vervollständigen Sie das Eingabewort und geben Sie die Reihenfolge der Spieler mit Ballbesitz an. 2 Punkte
- Turing-Kara ist eine Anwendung, welche das Spiel mit der oben angegebenen Symbolik simulieren kann. Öffnen Sie die Welt-Datei `16L21A.world`, welche bereits ein Eingabewort für die Beschreibung eines Spiels enthält. Über die Schaltfläche *Programmieren* gelangen Sie in den Programmeditor und öffnen dort die Programm-Datei `16L21A.kara`. Testen Sie die Funktion von Turing-Kara, indem Sie unter *Ausführen* das Wiedergabesymbol betätigen. Beobachten Sie die Schritte des Schreib-Lese-Kopfes (rotes Quadrat) und das Entstehen des Ausgabewortes. (Abbildungen umseitig)
Geben Sie Ihr Eingabewort aus Aufgabe b) in die Welt ein und testen Sie den Spielverlauf!
Stoppen Sie den Programmablauf an geeigneter Stelle und speichern Sie die entstandene Welt unter `Wanderball_c.world`. 2 Punkte

- d) Es wird nicht lange dauern, bis es Ärger gibt, weil einige Spieler der Runde den Ball nie erhalten werden. Begründen Sie diese Tatsache und geben Sie ein mögliches Eingabewort `richtung_d` sowie ein zugehöriges Ausgabewort `spiel_d` an, bei dem dieser Spielverlauf eintritt. 2 Punkte

Der Sportlehrer reagiert auf das Problem und legt als Spielregel fest: „Wer den Ball abgegeben hat, tauscht seine gemerkte Abspiel-Richtung.“

- e) Geben Sie zwei mögliche Eingabeworte `richtung_e1` und `richtung_e2` mit den zugehörigen Ausgabeworten `spiel_e1` und `spiel_e2` an, die ein Spiel mit 8 Ballwechseln nach der neuen Spielregel beschreiben. 2 Punkte

- f) Ändern Sie das Programm `16L21A.kara` so, dass es diese Regeln realisiert. Begründen Sie Ihre Änderungen. Speichern Sie das geänderte Programm unter `Wanderball_f.kara`
Legen Sie eine neue, leere Kara-Welt an und testen Sie Ihre Lösung mit den Eingabeworten aus Aufgabe e). Speichern Sie eine entstandene Welt unter `Wanderball_f.world`

4 Punkte

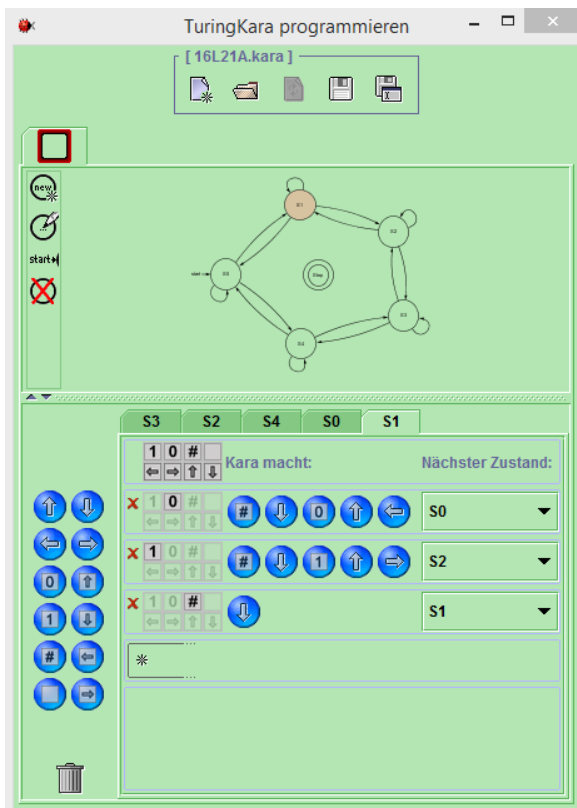


Abbildung 1: Turing-Kara, Spieler, Wurfrichtungen und Befehle

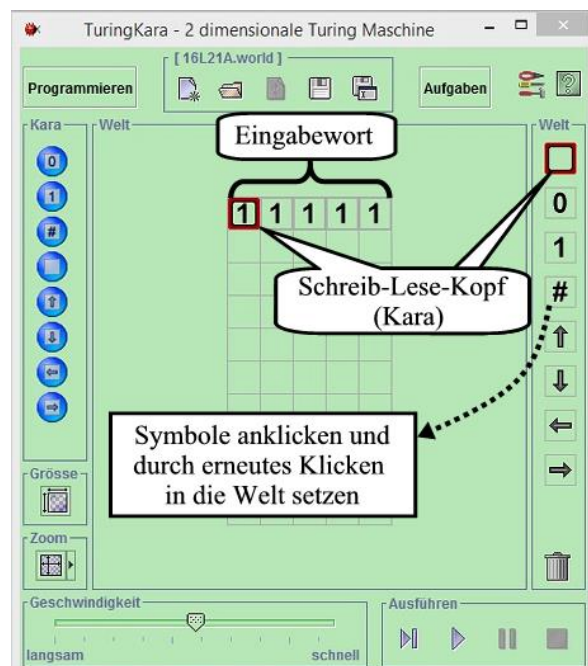


Abbildung 2: Turing-Kara - Welt mit Ein- und Ausgabeworten

6 Punkte

Klassenstufen 9/10

2. Aufgabe

Beide Verfahren benötigen für eine Bildfolge mit 10 Bildern $8 \cdot 8 \cdot 10 = 640$ Zeichen Speicherplatz. Diesen kann nach dem Prinzip der Lauflängencodierung einsparen: Dazu werden mehrfach nacheinander stehende Zeichen mit der Anzahl ihres Auftretens versehen. Der Code aus Aufgabe a) würde lauflängencodiert so aussehen:

10a3e4ae3ae3ae3ae4aeae6ae7ae12a

und demnach 31 Zeichen Speicherplatz belegen.

- f) Erweitern Sie Ihre Computerlösung derart, dass sowohl die eingegebenen Codes im Vollbildverfahren als auch die im Differenzbildverfahren lauflängencodiert ausgegeben werden. 5 Punkte

- g) Übernehmen Sie folgende Tabelle und füllen Sie diese aus. 2 Punkte

	animation_1.txt	animation_2.txt
Anzahl Zeichen Vollbildverfahren lauflängencodiert		
Anzahl Zeichen Differenzbildverfahren lauflängencodiert		

- h) Beschreiben Sie, welche Eigenschaft eine Bildfolge haben muss, damit die Kombination aus Differenzbildverfahren und Lauflängencodierung eine möglichst große Speicherplatzeinsparung ergibt. 2 Punkte

- i) Beurteilen Sie Vollbildverfahren und Differenzbildverfahren hinsichtlich Ihrer Toleranz bezüglich falsch abgespeicherter Werte einer Bildfolge. 2 Punkte