



Liebe Schülerin, lieber Schüler,

wir freuen uns, Sie heute zum Sächsischen Informatikwettbewerb 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 2 Stunden (120 min) Zeit.

Hilfsmittel

- Als Hilfsmittel sind Standardsoftware (Textverarbeitung, Tabellenkalkulation, Datenbanken) sowie Taschenrechner und Tafelwerk zugelassen.
- Über die zugelassenen Programmiersysteme informiert Sie Ihr Lehrer.

Bewertung

- Für die Aufgabe 1 gibt es 10 Punkte, für die Aufgabe 2 werden 20 Punkte vergeben.
- Zur Lösung erforderliche Algorithmen sollten zuerst in einer übersichtlichen Form (z.B. Struktogramm oder Programmablaufplan) angegeben werden. Die Programmidee ist aufzuschreiben und wird bewertet. Es ist wichtig, dass der Lösungsweg deutlich wird.
- Die Lösungsalgorithmen sollen möglichst effektiv sein.
- Bewertet wird auch ein guter Programmierstil. Diese Punkte werden erteilt, wenn der gefundene Algorithmus entsprechend der verwendeten Software umgesetzt wird. Dabei werden folgende Kriterien berücksichtigt:
 - bei Verwendung von Programmiersystemen
 - gut lesbare Form des Quelltextes
 - aussagekräftige Variablennamen
 - modulares Variablenkonzept
 - modulares Programmkonzept (Verwendung von Prozeduren und Funktionen)
 - nutzerfreundliche Ein- und Ausgabe
 - bei Verwendung von Tabellenkalkulationsprogrammen
 - geeignete Formatierung der Zellen
 - dem Inhalt entsprechende Gestaltung
 - sinnvolle Verwendung von Zellbezügen

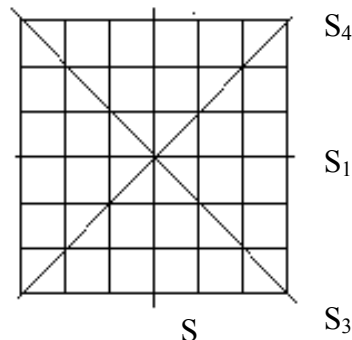
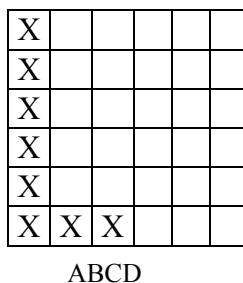


Regionalwettbewerb Klassenstufe 11/12

1. Aufgabe

L im Quadrat

Sie erinnern sich doch gern an Zeiten, in denen Sie sich mit Spiegelungen und Drehungen geometrischer Figuren in der Ebene beschäftigt haben. Als Schüler in der gymnasialen Oberstufe möchten Sie den Mitschülern unterer Klassen Freude an der Mathematik bereiten. Da bietet sich eine Computerlösung an, mit der man sich die Spiegelung an einer Geraden und Drehung um einen vorgegebenen Punkt experimentell erarbeiten kann. Die Zeit, ein solches Lernprodukt für beliebige geometrische Figuren zu realisieren, haben Sie heute leider nicht, können aber im Rahmen des Informatikwettbewerbes die Grundlagen dafür schaffen. Die Aufgabe wird daher auf die Spiegelung, Drehung und Invertierung einer Figur in einer 6x6-Rasterfläche eingeschränkt. Sie sehen unten zwei 6x6-Rasterflächen mit den Eckpunkten ABCD bzw. A'B'C'D'. Im linken Rasterfeld ist ein „L“ als Ausgangsfigur eingezeichnet. Dieses soll an den Geraden S1, S2, S3 und S4 gespiegelt und um den gemeinsamen Schnittpunkt der Geraden n-mal um 90° rechtsherum gedreht werden ($n \in \mathbb{N}$).



A'B'C'D'

Aufgaben:

- Zeichnen Sie in je ein 6x6-Rasterfeld das vorgegebene L nach fortlaufender Spiegelung an den Spiegelgeraden S1, S2, S3 und S4.
- Skizzieren Sie die Lage des „L“ nach 3 und 5 Drehungen um den Mittelpunkt.
- Erarbeiten Sie eine Computerlösung, die es gestattet mit Hilfe eines einfachen Editors eine Ausgangsfigur im 6x6-Raster zu erzeugen. Ein Menü soll folgende Manipulationen der Ausgangsfigur ermöglichen:
 - Spiegelung an einer der vorgegebenen Geraden.
 - Drehen um n-mal 90° um den Mittelpunkt des Rasterfeldes.
 - Invertieren der erzeugten Figur.

2. Aufgabe

Speditionsfirma

Eine Speditionsfirma beliefert immer von der Stadt A genau eine Firma in jeweils einer anderen Stadt und so weiter. Es ist immer der kürzeste Weg zwischen je 2 Städten bekannt. Nun müssen Touren zusammengestellt werden, die die Belieferung aller Städte vom Startpunkt A aus gewährleisten. Um kostengünstig zu arbeiten, muss nach Möglichkeit der kürzeste Weg insgesamt gewählt werden.



Aufgaben:

- Finden Sie für 5 Städte alle möglichen Fahrtrouten und stellen Sie diese auf dem Papier dar, wobei die Fahrtrichtung nicht beachtet werden muss. (z.B. ABCDEA = AEDCBA)
- Entwerfen Sie einen Algorithmus, der die möglichen Touren zusammenstellt und schreiben Sie eine Computerlösung.
- Schreiben Sie ein Programm, das dieses Problem mit maximal 10 Städten für eine Tour simuliert. Die Städte werden auf einer Matrix 10*10 zufällig angeordnet, die Anzahl der Städte wird zufällig ermittelt und in der Matrix verteilt. Die Länge einer möglichen Tour wird ermittelt.
- Entwickeln Sie einen Algorithmus, der die kürzeste Tour findet. Beschreiben Sie diesen Algorithmus und setzen Sie ihn in ihrem Programm um.