



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 2,5 Stunden (150 min) Zeit.

Hilfsmittel

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

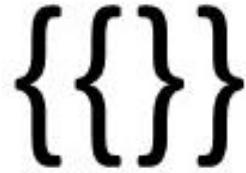
Bewertung

- Für die Aufgabe 1 gibt es 10 Punkte, für die Aufgabe 2 werden 20 Punkte vergeben.
- **Zu jeder Aufgabe ist ein Teil der Aufgaben auf dem Papier zu lösen. Beachten Sie dazu auch die Punktverteilung auf den Aufgabenzetteln.**
- Es ist wichtig, dass der Lösungsweg deutlich wird.

Wir wünschen Ihnen viel Erfolg!

Klammernchecker

Blöcke sind ein wichtiges Mittel der Programmstrukturierung. Sie werden in Programmiersprachen oft durch geschweifte Klammern gekennzeichnet. In der Praxis kommt es häufig vor, dass Klammern vergessen oder vertauscht werden. Dadurch entstehen inkorrekte Klammernfolgen, welche zu Fehlern beim Kompilieren führt.



In dieser Aufgabe sollen Sie eine Prüfung für korrektes Klammersetzen umsetzen.

Dazu entfernen wir aus einem Quelltext, der aus beliebigen Zeichen besteht, alle Zeichen, die keine geschweiften Klammern sind. Dadurch entsteht eine Folge von Klammern, die im folgenden Klammernfolge genannt wird.

Quelle: Angelehnt an IOM 2019 Aufgabenarchiv Informatik

Aufgaben:

- Geben Sie zwei inkorrekte Klammernfolgen mit jeweils mindestens 4 Klammern an und begründen Sie warum diese inkorrekt sind. 2 Punkte
- Erstellen Sie eine Computerlösung, die aus einem gegebenen Quelltext (Zeichenkette), die in ihm enthaltene Klammernfolge extrahiert und diese ausgibt. 2 Punkte
- Ein Algorithmus soll entscheiden, ob eine beliebige Klammernfolge korrekt ist. Beschreiben Sie eine Lösungsidee. 2 Punkte
- Implementieren Sie Ihren in c) beschriebenen Algorithmus. Es soll ausgegeben werden, ob eine einzugebende Klammernfolge korrekt ist. 2 Punkte

In manchen Fällen lässt sich eine inkorrekte Klammernfolge durch Verschieben von Klammern an andere Positionen in eine korrekte Folge verwandeln.

- Implementieren Sie einen Algorithmus, der entscheidet, ob es möglich ist, aus einer inkorrekten Klammernfolge eine korrekte zu machen, indem Klammern an andere Positionen verschoben werden. Ist die Klammernfolge korrigierbar, so soll der Algorithmus die Korrektur durchführen. Ausgegeben werden sollen die originale inkorrekte Klammernfolge, die Aussage „nicht korrigierbar“ oder „korrigierbar“ und gegebenenfalls die korrigierte Klammernfolge. 2 Punkte

Besondere Zahlen

In der Mathematik beschäftigt man sich gern mit Zahlen, die besondere Eigenschaften haben. Am bekanntesten sind sicher die Primzahlen, aber es gibt auch perfekte und superperfekte Zahlen, fröhliche und traurige Zahlen, potente Zahlen, Vampirzahlen und viele andere.

Eine **superperfekte Zahl** ist halb so groß wie die Summe der Teiler der Summe ihrer Teiler.

Zum Beispiel ist 16 eine superperfekte Zahl, denn

- 16 hat die Teiler 1, 2, 4, 8 und 16, deren Summe ist 31
- 31 hat die Teiler 1 und 31, deren Summe ist $32 = 2 \cdot 16$

Eine **Vampirzahl** ist eine Zahl mit gerader Anzahl ($2n$) von Ziffern, aus denen sich zwei n -stellige Zahlen bilden lassen, deren Produkt die Ausgangszahl ist. Die Faktoren dürfen nicht beide mit einer Null enden.

Zum Beispiel ist 1530 eine Vampirzahl, denn $1530 = 51 \cdot 30$

Aufgaben:

- Begründen Sie, dass 1395 eine Vampirzahl ist und dass 64 eine superperfekte Zahl ist.
2 Punkte
- Entwickeln Sie eine Computerlösung, mit der mindestens die ersten sieben superperfekten Zahlen berechnet werden können und geben Sie diese Zahlen an. 5 Punkte
- Äußern Sie eine Vermutung über besondere Eigenschaften weiterer superperfekter Zahlen. (Übrigens ist bis heute nicht klar, ob es auch ungerade superperfekte Zahlen gibt.)
1 Punkt
- Beschreiben Sie einen Algorithmus, mit dem sich bei der Suche nach Vampirzahlen zu einer gegebenen $2n$ -stelligen Zahl alle Paare von n -stelligen Zahlen bilden lassen. Setzen Sie diesen Algorithmus in eine Computerlösung um.
6 Punkte
- Entwickeln Sie eine Computerlösung, die alle vierstelligen Vampirzahlen bestimmt und ausgibt.
3 Punkte
- Ändern Sie Ihre Computerlösung aus Aufgabe e) so um, dass sie (wenigstens) eine sechsstellige Vampirzahl findet, für die es mehr als eine Möglichkeit zur Bildung der beiden Faktoren gibt. Geben Sie eine solche Zahl an.
3 Punkte