

Dies ist die erste Runde der Schweizer Informatik-Olympiade. Der Test besteht aus 16 Multiple-Choice-Fragen, die jeweils 1 Punkt wert sind.

Regeln: Diese Prüfung muss alleine und ohne KI absolviert werden. Taschenrechner sind erlaubt, aber nicht erforderlich.

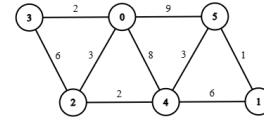
Allgemeine Fragen (10 Fragen): Enthält Fragen zum logischen Denken und zu einfachen informatikbezogenen Konzepten

Programmfragen (6 Fragen): Enthält Fragen mit leicht verständlichem Pseudocode und einige Fragen dazu. Vorkenntnisse in Programmierung sind nicht erforderlich.

Wichtig: Die Aufgaben in jedem Abschnitt sind nach Schwierigkeitsgrad sortiert. Denk daran, Aufgaben aus beiden Abschnitten zu versuchen.

Zeitlimit: 40 Minuten

Zeitspanne: 40 minutes



A. 10 B. 11 C. 12 D. 13 E. 14

1 Allgemeine Fragen

Dies sind die allgemeinen Fragen. Die folgenden 10 Fragen sind nach Schwierigkeitsgrad geordnet. Alle Fragen sind 1 Punkt wert. Für falsche Antworten gibt es keine Punktabzüge.

Frage 2

Gegeben sei die Folge $[5, 2, 1, 3]$. Wie viele Schritte sind mindestens nötig, um die Folge in aufsteigender Reihenfolge zu sortieren? In jedem Schritt darfst du zwei benachbarte Elemente vertauschen. Zum Beispiel kann $[5, 2, 1, 3]$ in einem Schritt zu $[5, 2, 3, 1]$ werden, aber nicht in einem Schritt zu $[1, 2, 5, 3]$.

A. 3 B. 4 C. 5 D. 6 E. 7

Frage 3

Gegeben seien 6 Städte (nummeriert 0, 1, 2, 3, 4, 5), die durch Strassen miteinander verbunden sind, welche in beide Richtungen durchquert werden können. Jede Strasse verbindet genau zwei Städte und benötigt eine bestimmte Anzahl von Minuten für die Durchquerung, wie im obigen Diagramm angegeben. Wie viele Minuten benötigt man mindestens, um von Stadt 3 nach Stadt 1 zu gelangen?

Frage 4

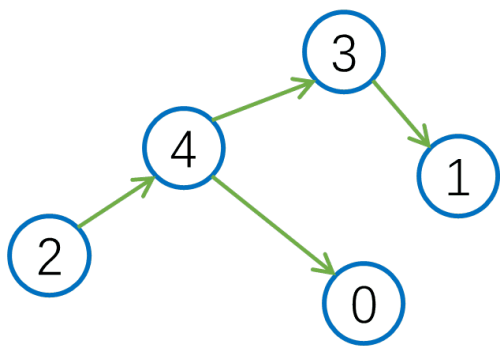
Du hast 9 gelbe Enten und 99 weisse Enten. Du kannst eine zusätzliche gelbe Ente für 1 CHF kaufen und eine weisse Ente ebenfalls für 1 CHF. Jede Ente darfst du kostenlos rot anmalen. Wie viele CHF musst du mindestens bezahlen, damit du am Ende gleich viele Enten jeder Farbe hast, wobei du keine Enten loswerden darfst?

A. 39 CHF B. 42 CHF C. 69 CHF D. 90 CHF E. 189 CHF

Frage 5

Gegeben seien 5 Städte (nummeriert 0, 1, 2, 3, 4), die durch Einbahnstrassen verbunden sind. Die Karte oben zeigt die derzeitigen Strassen zwischen den Städten. Mit den bereits eingezeichneten Strassen allein ist es nicht möglich, von einer Stadt zu starten und dabei jede andere Stadt genau einmal zu besuchen.

Welche der folgenden neuen Strassen würde — wenn sie zur Karte hinzugefügt wird — erlauben, alle Städte genau einmal zu besuchen, ausgehend von einer Stadt und den Richtungen der folgenden Strassen?



- A. $0 \rightarrow 1$ B. $0 \rightarrow 3$ C. $2 \rightarrow 0$ D. $2 \rightarrow 3$
 E. $4 \rightarrow 1$

Frage 6

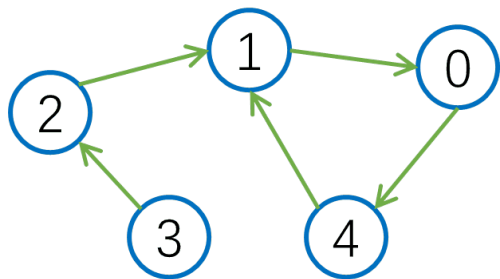
Maus Stoff hat eine Zahl zwischen 1 und 25 gewählt, welche er dir nicht verrät. Du darfst jeweils einen zusammenhängenden Zahlenbereich raten (z. B. $4 - 7$). Maus Stoff sagt dir dann, ob seine Zahl in deinem Bereich liegt, oder ob sie grösser bzw. kleiner ist. Wie oft musst du mindestens raten, damit du Stoffs Zahl mit Sicherheit herausfinden kannst?

- A. 3 B. 4 C. 5 D. 6 E. 7

Frage 7

Gegeben seien 5 Städte (nummeriert 0, 1, 2, 3, 4), die durch Einbahnstrassen verbunden sind. Du darfst genau eine neue Einbahnstrasse zwischen zwei Städten hinzufügen, zwischen denen bisher noch keine Strasse existiert (insbesondere darf keine Strasse hinzugefügt werden, wenn bereits eine Strasse in die entgegengesetzte Richtung existiert).

Auf wie viele verschiedene Arten kannst du eine solche Strasse hinzufügen, sodass anschliessend jede Stadt von jeder anderen Stadt aus erreichbar ist, wenn man den Richtungen der Einbahnstrassen folgt?



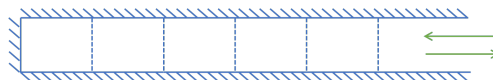
- A. 0 B. 1 C. 2 D. 3 E. 4

Frage 8

Es gibt einen schmalen Parkplatz mit 5 nebeneinanderliegenden Stellplätzen in einer Reihe. Insbesondere kann ein Auto nur ausparken, wenn zwischen ihm und der Ausfahrt kein anderes Auto steht. Die fünf Autos A, B, C, D und E fahren nacheinander auf den Parkplatz (A zuerst, dann B , dann C , dann D , dann E).

Hinweis: Autos dürfen den Parkplatz verlassen, bevor alle fünf eingetroffen sind.

Welche der folgenden Ausparkreihenfolgen ist nicht möglich?



- A. A, B, C, D, E B. B, A, D, C, E
 C. C, B, E, D, A D. E, D, C, B, A
 E. B, C, E, A, D

Frage 9

Gegeben sei eine Liste aus 4 verschiedenen Zahlen. Du darfst immer nur zwei Zahlen miteinander vergleichen, um festzustellen, welche grösser ist. Unabhängig von der anfänglichen Reihenfolge: Wie viele Vergleiche sind mindestens erforderlich, um garantiert alle Zahlen in aufsteigender Reihenfolge sortieren zu können?

- A. 3 B. 4 C. 5 D. 6 E. 7

Frage 10

Das Land Crazystoff besteht aus 2026 Inseln, von 0 bis 2025 durchnummeriert. Es ist bekannt, dass es von jeder Insel x eine beidseitige Flugverbindung (Hin- und Rückflug) zur Insel $\lfloor \frac{x}{2} \rfloor$ gibt. Wie viele Flüge muss man mindestens nehmen, um von Insel 125 zur Insel 1025 zu gelangen?

Hinweis: $\lfloor y \rfloor$ ist die grösste Ganzzahl, die den Wert y nicht überschreitet. Zum Beispiel $\lfloor 3.14 \rfloor = 3$, $\lfloor 7 \rfloor = 7$, $\lfloor \frac{11}{2} \rfloor = 5$.

- A. 10 B. 11 C. 12 D. 14 E. 16

Frage 11

Wie viele Möglichkeiten gibt es, ein 4×14 -Raster vollständig mit 1×4 -Kacheln zu bedecken, sodass sich keine Kacheln überlappen? Hinweis: Die Kacheln dürfen gedreht werden.

- A. 43 B. 44 C. 50 D. 55 E. 59

2 Programmfragen

Dies sind die Programmfragen. Die folgenden 6 Fragen sind nach Schwierigkeitsgrad geordnet. Alle Fragen sind mit 1 Punkt bewertet. Für falsche Antworten gibt es keine Punktabzüge.

Frage 12

Was ist die Ausgabe des folgenden Programms?

Legende

- **Set** weist einer Variable einen Wert zu.
- **If**, **Else If**, or **Else** sind Bedingungen. Sie führen einen bestimmten Codeblock aus, wenn die Bedingung (die in derselben Zeile angegeben ist) wahr ist. Einrückungen zeigen an, welche Codezeilen zu jeder Bedingung gehören. Wenn keine der Bedingungen erfüllt ist, wird der Codeblock unter **Else** ausgeführt.
- **Print** gibt den Wert einer Variablen aus.
- **Print** gibt den Wert einer Variable aus.

```
Set originalScore ← 76
Set bonus ← 0

If originalScore ≥ 90
  Set bonus ← 5
Else If originalScore ≥ 75
  Set bonus ← 2
Else
  Set bonus ← 0

Set finalScore ← originalScore + bonus
Print finalScore
```

- A. 74 B. 76 C. 78 D. 81 E. 83

Frage 13

Was ist die Ausgabe des folgenden Programms?

Legende

- Eine **For**-Schleife wiederholt den Code eine festgelegte Anzahl von Malen, wie durch den Bereich einschliesslich angegeben. Beispielsweise würde **For x from 2 to 4** den Code dreimal durchlaufen. Zuerst für $x = 2$, dann für $x = 3$ und schliesslich für $x = 4$.

```
Set sum ← 0
For i from 1 to 3
  Set sum ← sum + i * i
Print sum
```

- A. 3 B. 6 C. 9 D. 14 E. 27

Frage 14

Was ist die Ausgabe des folgenden Programms?

Legende

- Eine **While**-Schleife wird so lange ausgeführt, wie die Bedingung wahr bleibt. Die Schleife überprüft ihre Bedingung zu Beginn jeder Iteration. Sobald die Bedingung falsch wird, wird die Schleife beendet und das Programm fährt mit der nächsten Anweisung nach der Schleife fort.
- $x == y$ ist wahr, wenn die Werte von x und y gleich sind, und ist in allen anderen Fällen falsch. Beispielsweise ist $1 == 2$ falsch und $5 == 5$ wahr.

```
Set N ← 2026
Set M ← 1

while (M == 1)
  If N < 0
    Set M ← 0

  Set N ← N - 2

Print N
```

- A. -4 B. -3 C. -2 D. -1 E. 0

Frage 15

Was ist die Ausgabe des folgenden Programms?

Legende

- **Function** definiert einen Codeabschnitt, der mehrmals aufgerufen werden kann. Er enthält in der Regel einen Eingabewert und einen Rückgabewert.
- $a \bmod b$ gibt den Rest zurück, wenn a durch b geteilt wird, z. B. $7 \bmod 3 = 1$ und $8 \bmod 4 = 0$.

```
Function foo(x)
  If x mod 50 == 0
    Return x / 50
  Else
    Return foo(x - 1) * foo(x - 1)
End Function

Print foo(99) + foo(100) + foo(101)
```

- A. 0 B. 7 C. $2^{2^{48}} + 6$ D. $2^{49} + 2^{50} + 2^{51}$
 E. Keine der oben genannten Antworten

Frage 16

Was ist die Ausgabe des folgenden Programms?

Legende

- **Indices** sind Positionsnummern, die auf die Position eines Elements in einer Liste verweisen. Es wird bei 0 angefangen zu zählen. Wenn beispielsweise $\text{list} = (2, 4, 6)$ ist, dann ist $\text{list}[0] = 2$ und $\text{list}[2] = 6$.

```
Function CountSafeSquares(n, m, rooks)
  Set count ← 0
  Set rookCount ← length of rooks

  For i from 0 to n - 1
    For j from 0 to m - 1
      Set safe ← true

      For k from 0 to rookCount - 1
        Set rx ← rooks[k][0]
        Set ry ← rooks[k][1]

        If i == rx or j == ry
          Set safe ← false
          Break

      If safe == true
        Set count ← count + 1

  Return count
End Function

Print CountSafeSquare(10, 15, [(2, 3), (5, 7), (5, 3)])
```

- A. 84 B. 104 C. 112 D. 126 E. 150

Frage 17

Wie viele verschiedene Werte kann K anfangs annehmen (gelbes Feld), sodass unser Programm für ein beliebig gewähltes N (gelbes Feld) „Yay!“ ausgibt?

Anmerkung: N und K müssen ganze Zahlen sein.

```
Set M ← 7
Set N ←       
Set K ←       

If (K < 0)
  Set K ← 0

While (K < 6)
  Set M ← M + N
  Set K ← K + 1

If (M == 42) Print "Yay!"
```

- A. 0 B. 1 C. 2 D. 3 E. Unendlich
 viele mögliche Werte