

Wettbewerb Mathe-Info - März 2022
Schriftliche Prüfung in Informatik

WICHTIG ZU BEACHTEN:

Wenn keine anderen Spezifikationen vorliegen, wird davon ausgegangen, dass alle arithmetischen Operationen für unbegrenzte Datentypen ausgeführt werden (es gibt keinen Überlauf / Unterlauf).

Man nimmt an, dass die Indizierung aller Arrays/Sequenzen bei 1 beginnt.

1. Wir betrachten den Algorithmus `magic(x)`, wobei x eine natürliche Zahl ist ($1 \leq x \leq 32000$).

```
Algorithm magic(x):
  st ← 1
  dr ← x
  While st ≤ dr execute
    mj ← (st + dr) DIV 2
    If mj * mj = x then
      return True
    EndIf
    If mj * mj < x then
      st ← mj + 1
    else
      dr ← mj - 1
    EndIf
  EndWhile
  return False
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

- A. Der Algorithmus gibt für alle Eingabewerte x den Wert *False* zurück, wenn $x < 10$ ist.
- B. Der Algorithmus zerlegt die Zahl x in ihre Primfaktoren.
- C. Der Algorithmus prüft, ob die Zahl x eine perfekte Quadratzahl ist.
- D. Der Algorithmus gibt für keinen gültigen Wert des Eingabeparameters x *True* zurück.

2. Wir betrachten den Algorithmus `calculeaza(a, b)`, wobei a und b natürliche Zahlen sind ($1 \leq a, b \leq 10000$).

```
Algorithm calculeaza(a, b):
  x ← 1
  For i ← 1, b execute
    x ← (x MOD 10) * a
  EndFor
  return x
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

- A. Wenn $a = 107$ und $b = 101$ ist, liefert der Algorithmus den Wert 107.
- B. Wenn $a = 1001$ und $b = 101$ ist, liefert der Algorithmus den Wert 1001.
- C. Bei allen Aufrufen des Algorithmus mit $1 \leq a \leq 10000$ und $b = 101$ ist der zurückgegebene Wert der Wert von a .
- D. Für alle Algorithmus-Aufrufe mit $a = 1001$ und $1 \leq b \leq 10000$ ist der zurückgegebene Wert 1001.

3. Betrachten wir den Algorithmus $\text{afis}(n)$, wobei n eine natürliche Zahl ist ($0 \leq n \leq 10000$).

```
Algorithm afis(n):
  Write n, " "
  If n > 0 then
    afis(n DIV 2)
    Write n, ", "
  EndIf
EndAlgorithm
```

Was wird nach dem Aufruf von $\text{afis}(n)$ angezeigt?

- A. Es wird eine Folge von Zahlen angezeigt, wobei das erste Element gleich dem letzten ist, das zweite dem vorletzten usw. (außer dem mittleren Element).
- B. Es wird eine Folge von geraden Zahlen angezeigt.
- C. Es wird eine Folge von Zahlen in aufsteigender Reihenfolge, gefolgt von Zahlen in absteigender Reihenfolge angezeigt.
- D. Es wird eine Folge von Zahlen in absteigender Reihenfolge, gefolgt von Zahlen in aufsteigender Reihenfolge angezeigt.

4. Wir betrachten den Algorithmus $\text{cauta}(n, b)$, wobei n und b natürliche Zahlen sind ($0 \leq n \leq 10^6$, $2 \leq b < 10$).

```
Algorithm cauta(n, b):
  v ← 0
  If n = 0 then
    return 1
  else
    m ← n
    While m > 0 execute
      If m MOD b = 0 then
        v ← v + 1
      EndIf
      m ← m DIV b
    EndWhile
    return v
  EndIf
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

- A. Der Algorithmus bestimmt, wie viele Ziffern die Zahl n hat, und gibt sie zurück.
- B. Der Algorithmus gibt 1 zurück, wenn die Zahl n eine Potenz von b ist, und sonst 0.
- C. Der Algorithmus bestimmt die Anzahl der Ziffern 0 in der Darstellung der Zahl n zur Basis b und gibt sie zurück.
- D. Der Algorithmus gibt 1 zurück, wenn die Zahl n mit der Ziffer b endet, und sonst 0.

5. Wir betrachten den Algorithmus $\text{abc}(a, n, p)$, wobei n eine natürliche Zahl ($1 \leq n \leq 10000$), p eine ganze Zahl ($-10000 \leq p \leq 10000$) und a eine Folge von n natürlichen Zahlen ($a[1], a[2], \dots, a[n]$) ist, die nicht Null sind.

```
Algorithm abc(a, n, p):
  If n < 1 then
    return -1
  else
    If (1 ≤ p) AND (p ≤ n) then
      return a[p]
    else
      return 0
    EndIf
  EndIf
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

- A. Der Algorithmus gibt 0 zurück genau dann, wenn p negativ oder größer als n ist.
- B. Der Algorithmus gibt das Element an der Position p zurück, wenn p strikt größer als 0 und kleiner oder gleich der Länge der Zeichenkette ist.
- C. Der Algorithmus gibt niemals 0 für Parameterwerte zurück, die die Vorbedingungen der Anweisung erfüllen.
- D. Der Algorithmus gibt das Element an der Position p zurück, wenn p größer oder gleich 0 und kleiner als die Länge der Zeichenkette ist

6. Um n -stellige Zahlen zu erzeugen, die nur aus den Ziffern 0, 6, 7 bestehen, wird ein Algorithmus verwendet, der für $n = 2$ die Zahlen 60, 66, 67, 70, 76, 77 in aufsteigender Reihenfolge erzeugt.

Wenn $n = 4$ und derselbe Algorithmus verwendet wird, wie lautet die Zahl, die unmittelbar nach der Zahl 6767 erzeugt wird?

- A. 7667
- B. 6760
- C. 6776
- D. Keine der anderen Optionen

7. Für eine natürliche Zahl nr ($1000 \leq nr \leq 1000000$) definieren wir die Dekrementierungsoperation wie folgt: Wenn die letzte Ziffer von nr nicht 0 ist, subtrahieren wir 1 von nr , andernfalls teilen wir nr durch 10 und behalten nur den ganzzahligen Teil. Welcher der folgenden Algorithmen gibt bei dem Aufruf `decrementare(nr, k)` die Zahl zurück, die man erhält, wenn man k -mal ($0 \leq k \leq 100$) die Dekrementierungsoperation auf die Zahl nr anwendet? Für $nr = 15243$ und $k = 10$ ist das Ergebnis beispielsweise 151.

A.

```
Algorithm decrementare(nr, k):
  If k = 0 then
    return nr
  else
    If nr MOD 10 ≠ 0 then
      return decrementare(nr - 1, k - 1)
    Else
      return decrementare(nr DIV 10, k - 1)
    EndIf
  EndIf
EndAlgorithm
```

B.

```
Algorithm decrementare(nr, k):
  While k > 0 execute
    If nr MOD 10 = 0 then
      nr ← nr DIV 10
    else
      nr ← nr - 1
    EndIf
    k ← k - 1
  EndWhile
  return nr
EndAlgorithm
```

C.

```
Algorithm decrementare(nr, k):
  For i ← 1, k execute
    If nr MOD 10 > 0 then
      nr ← nr DIV 10
    else
      nr ← nr - 1
    EndIf
  EndFor
  return nr
EndAlgorithm
```

D.

```
Algorithm decrementare(nr, k):
  If k = 0 then
    return nr
  else
    If k > nr MOD 10 then
      nr1 ← nr DIV 10
      return decrementare(nr1, k - nr MOD 10 - 1)
    else
      return decrementare(nr - k, 0)
    EndIf
  EndIf
EndAlgorithm
```

8. Der folgende Algorithmus hat als Eingabeparameter einen Vektor v mit n natürlichen Zahlen ($v[1]$, $v[2]$, ..., $v[n]$) und die ganze Zahl n ($1 \leq n \leq 10000$).

```
Algorithm fn(v, n):
  a ← 0
  For i ← 1, n execute
    ok ← True
    b ← v[i]
    While (b ≠ 0) AND (ok = True) execute
      If b MOD 2 = 1 then
        ok ← False
      EndIf
      b ← b DIV 10
    EndWhile
    If ok = True then
      a ← a + 1
    EndIf
  EndFor
  return a
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

- A. Der Algorithmus gibt die Anzahl der ungeraden Elemente im Vektor v zurück.
- B. Der Algorithmus gibt die Anzahl der Elemente im Vektor v zurück, die Potenzen von 2 sind.
- C. Der Algorithmus gibt die Anzahl der Elemente im Vektor v zurück, die nur gerade Ziffern haben.
- D. Der Algorithmus gibt die Anzahl der Elemente im Vektor v zurück, die nur ungerade Ziffern in ihrer Zusammensetzung haben.

9. Der Algorithmus `magic(s, n)` hat als Eingabeparameter eine Zeichenkette s mit n Zeichen ($s[1]$, $s[2]$, ..., $s[n]$) und die ganze Zahl n ($1 \leq n \leq 10000$).

```
Algorithm magic(s, n):
  i ← n
  While 1 ≤ i execute
    If s[i] ≠ s[n - i + 1] then
      return 0
    EndIf
    i ← i - 1
  EndWhile
  return 1
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

- A. Der Algorithmus gibt 1 zurück, wenn s eine gerade Anzahl von Zeichen hat.
- B. Der Algorithmus gibt 1 zurück, wenn s ein Palindrom ist.
- C. Der Algorithmus hat einen Fehler, weil der Ausdruck $n - i + 1$ während der Ausführung negative Werte annehmen kann.
- D. Der Algorithmus gibt 1 zurück, wenn s nur alphanumerische Zeichen enthält.

10. Betrachten Sie die folgende Codesequenz in Pseudocode:

```
Read a
For i ← 1, a - 1 execute
  For j ← i + 2, a execute
    If i + j > a - 1 then
      Write a, " ", i, " ", j
      Write new line
    EndIf
  EndFor
EndFor
```

Wie viele Lösungspaare werden nach Ausführung der Codesequenz für $a = 9$ angezeigt?

- A. 13
- B. 15
- C. 19
- D. keine der anderen Antworten ist richtig

11. Der Algorithmus $ceFace(n)$ hat als Eingabeparameter eine natürliche Zahl n ($0 \leq n \leq 10000$).

```
Algorithm ceFace(n):
  s ← 0
  While n > 0 execute
    c ← n MOD 10
    If c MOD 2 = 0 then
      s ← s + c
    EndIf
    n ← n DIV 10
  EndWhile
  return s
EndAlgorithm
```

Was gibt der Aufruf $ceFace(9876)$ zurück?

- A. 16
- B. 48
- C. 14
- D. 63

12. Der Algorithmus $generation(n)$ verarbeitet eine natürliche Zahl n ($0 < n < 100$).

```
Algorithm generare(n):
  nr ← 0
  For i ← 1, 1801 execute
    used[i] ← False
  EndFor
  While not used[n] execute
    sum ← 0
    used[n] ← True
    While n ≠ 0 execute
      digit ← n MOD 10
      n ← n DIV 10
      sum ← sum + digit * digit * digit
    EndWhile
    n ← sum
    nr ← nr + 1
  EndWhile
  return nr
EndAlgorithm
```

Welche der folgenden Aussagen sind zutreffend?

- A. Wenn $n = 10$ ist, wird der Wert 2 zurückgegeben.
- B. Wenn $n = 10$ ist, wird der Wert 1 zurückgegeben.
- C. Wenn $n = 3$ ist, wird der Wert 4 zurückgegeben.
- D. Die beiden Aufrufe `generate(3)` und `generate(30)` geben denselben Wert zurück.

13. Wir betrachten den Algorithmus $f(a, b)$, wobei a und b natürliche Zahlen sind ($1 \leq a < b \leq 1000$):

```
Algorithm f(a, b):
  If a > 0 then
    return b + f(a DIV 2, b * 2)
  EndIf
  return b + f(a * 2, b DIV 2)
EndAlgorithm
```

Unglücklicherweise ruft sich der Algorithmus unendlich oft selbst auf. Geben Sie an, welchen Wert der Parameter b haben wird, wenn der Parameter a zum ersten Mal 0 wird. Der Algorithmus wird mit der Anweisung

$c \leftarrow f(20, 10)$ aufgerufen.

- A. 100
- B. 160
- C. 320
- D. 640

14. Geben Sie an, welcher der folgenden Ausdrücke nur dann wahr ist, wenn die natürliche Zahl n durch 3 teilbar ist und die letzte Ziffer 4 oder 6 ist:

- A. $(n \text{ MOD } 3 = 0) \text{ OR } ((n \text{ MOD } 10 = 4) \text{ AND } (n \text{ MOD } 10 = 6))$
- B. $(n \text{ MOD } 6 = 0) \text{ AND } ((n \text{ MOD } 10 = 4) \text{ OR } (n \text{ MOD } 10 = 6))$
- C. $((n \text{ MOD } 9 = 0) \text{ AND } (n \text{ MOD } 10 = 4)) \text{ OR } ((n \text{ MOD } 3 = 0) \text{ AND } (n \text{ MOD } 10 = 6))$
- D. $(n \text{ MOD } 3 = 0) \text{ AND } (((n \text{ MOD } 2 = 0) \text{ AND } (n \text{ MOD } 5 = 0)) \text{ OR } ((n \text{ MOD } 2 = 0) \text{ AND } (n \text{ MOD } 5 = 1)))$

15. Betrachten Sie den folgenden logischen Ausdruck $(X \text{ OR } Z) \text{ AND } (X \text{ OR } Y)$. Wählen Sie die Werte für X, Y, Z so, dass die Auswertung des Ausdrucks das Ergebnis *Wahr* ergibt:

- A. $X \leftarrow \text{False}; Y \leftarrow \text{False}; Z \leftarrow \text{True};$
- B. $X \leftarrow \text{True}; Y \leftarrow \text{False}; Z \leftarrow \text{False};$
- C. $X \leftarrow \text{False}; Y \leftarrow \text{True}; Z \leftarrow \text{False};$
- D. $X \leftarrow \text{True}; Y \leftarrow \text{True}; Z \leftarrow \text{True};$

16. Betrachten Sie alle Zeichenketten der Länge $l \in \{1, 2, 3\}$, die aus Buchstaben der Menge $\{a, b, c, d, e\}$ bestehen. Wie viele dieser Zeichenfolgen haben streng aufsteigende Elemente (dem Alphabet entsprechend) und eine ungerade Anzahl von Konsonanten? (b, c und d sind Konsonanten)

- A. 14
- B. 13
- C. 26
- D. 81

17. Sie möchten ein Quadrat mit seinen Diagonalen anzeigen, indem Sie nur die Zeichen * (Sternchen) und . (Punkt) (für den Raum innerhalb des Quadrats außer den Diagonalen) verwenden. Das folgende Beispiel zeigt ein Quadrat mit den Seiten $n = 6$ Sternchen. Hierfür war es notwendig, 28 Sternchen und 8 Punkte zu verwenden.

```

* * * * *
* * . . * *
* . * * . *
* . * * . *
* * . . * *
* * * * *

```

Welche der folgenden Aussagen sind zutreffend?

- A. Für $n = 5$ werden genau 22 Sternchen und 4 Punkte benötigt.
- B. Für $n = 7$ werden genau 34 Sternchen und 15 Punkte benötigt.
- C. Für $n = 7$ werden genau 33 Sternchen und 16 Punkte benötigt.
- D. Für $n = 18$ werden genau 100 Sternchen und 224 Punkte benötigt.

18. Wir betrachten den Algorithmus `ceFace(T, n, e)`, der als Parameter eine Zeichenkette T mit n natürlichen Zahlen in aufsteigender Reihenfolge ($T[1], T[2], \dots, T[n]$) und die natürlichen Zahlen n und e ($1 \leq n, e \leq 10000$) annimmt.

```

Algorithm ceFace(T, n, e):
  If e MOD 2 = 0 then
    a ← 1
    b ← n
    While a ≤ b execute
      m ← (a + b) DIV 2
      If e < T[m] then
        b ← m - 1
      else
        If e > T[m] then
          a ← m + 1
        else
          return m
        EndIf
      EndIf
    EndWhile
    return 0
  else
    c ← 1
    g ← 0
    While (c ≤ n) AND (g = 0) execute
      If e = T[c] then
        g = c
      EndIf
      c ← c + 1
    EndWhile
    return g
  EndIf
EndAlgorithm

```

Welche der folgenden Aussagen sind zutreffend?

- A. Der Algorithmus gibt 0 zurück, wenn die Zahl e nicht in der Zeichenkette T enthalten ist.
- B. Wenn die Zahl e ungerade ist und in der Zeichenkette T vorkommt, gibt der Algorithmus die Position in der Zeichenkette T zurück, an der sich e befindet, mit Hilfe des binären Suchalgorithmus.
- C. Wenn die Zahl e ungerade ist und in der Zeichenkette T vorkommt, gibt der Algorithmus die Position in der Zeichenkette T an, an der sich e befindet, indem er den Algorithmus der sequentiellen Suche verwendet.
- D. Der Algorithmus gibt die Position in der Zeichenkette T zurück, an der sich die Zahl e befindet.

19. Wir betrachten den Algorithmus $\text{calcul}(x, n)$, wobei die Eingabeparameter die natürlichen Zahlen n und x sind, mit der Bedingung $1 \leq x \leq n < 10$.

```

Algorithm calcul(x, n):
  b ← 1
  For i ← 1, n - x execute
    b ← b + i
  EndFor
  a ← b
  For i ← n - x + 1, n execute
    a ← a + i
  EndFor
  return a - b
EndAlgorithm

```

Welche der folgenden Aussagen sind zutreffend?

- A. Wenn $n = 5$ und $x = 2$ ist, liefert der Algorithmus den Wert 20.
- B. Wenn $n = 3$ und $x = 2$ ist, liefert der Algorithmus den Wert 5.
- C. Der Algorithmus liefert die Kardinalität der Menge $\{\overline{c_1 c_2 \dots c_x} : c_i \neq c_j \forall 1 \leq i, j \leq x, i \neq j, 1 \leq c_i \leq n\}$
- D. Der Algorithmus liefert immer einen Wert, der streng größer als 0 ist.

20. Wir betrachten den Algorithmus $s(a, b, c)$, wobei a, b, c positive natürliche Zahlen sind ($1 \leq a, b, c \leq 10000$).

```

Algorithm s(a, b, c):
  If (a = 1) OR (b = 1) OR (c = 1) then
    return 1
  else
    If a > b then
      return a * s(a - 1, b, c)
    else
      If a < b then
        return b * s(a, b - 1, c)
      else
        return c * s(a - 1, b - 1, c - 1)
      EndIf
    EndIf
  EndIf
EndAlgorithm

```

20a. Geben Sie an, welche der folgenden Aussagen wahr sind, wenn $a = b$ und $a < c$ ist:

- A. Der Algorithmus berechnet und liefert $c! / (c - a)!$
- B. Der Algorithmus berechnet und liefert $c! / (c - a + 1)!$
- C. Der Algorithmus berechnet und liefert $c! / (c - a - 1)!$
- D. Der Algorithmus berechnet und liefert den Binomialkoeffizienten c über $(a - 1)$, also C_c^{a-1}

20b. Für $a = 3, b = 4, c = 7$, gibt der Algorithmus folgender Wert zurück:

- A. 224
- B. 56
- C. 336
- D. 168

21. Wir betrachten den Algorithmus $\text{numere}(a, b, c, d, e)$, der als Parameter fünf ganze Zahlen a, b, c, d und e benötigt ($1 \leq a, b \leq 10000, 2 \leq c \leq 16, 1 \leq d < c$).

```

Algorithm numere(a, b, c, d, e):
  If a = 0 AND b = 0 then
    If e = 0 then
      return True
    else
      return False
  EndIf
  EndIf
  If a MOD c = d then
    e = e + 1
  EndIf
  If b MOD c = d then
    e = e - 1
  EndIf
  return numere(a DIV c, b DIV c, c, d, e)
EndAlgorithm

```

Geben Sie an, welche der folgenden Aussagen für den Aufruf $\text{numere}(a, b, c, d, 0)$ zutreffen:

- A. Der Algorithmus liefert *True*, wenn die c -Basis-Darstellungen der Zahlen a und b die Ziffer d gleich oft enthalten, andernfalls *False*
- B. Der Algorithmus gibt *True* zurück, wenn die Ziffer d in der c -Basis-Darstellung der Zahl a und in der c -Basis-Darstellung der Zahl b erscheint, andernfalls *False*
- C. Der Aufruf $\text{numere}(a, b, c, d, 0)$ liefert denselben Wert wie der Aufruf $\text{numere}(b, a, c, d, 0)$
- D. Der Algorithmus gibt *True* zurück, wenn die Ziffer d an den gleichen Stellen in der c -Basis-Darstellung der Zahlen a und b erscheint, andernfalls *False*

22. Sei s eine Folge von natürlichen Zahlen, wobei die Elemente s_i der Form

$$s_i = \begin{cases} x, & \text{falls } i = 1 \\ x + 1, & \text{falls } i = 2 \\ s_{(i-1)}@s_{(i-2)} & \text{falls } i > 2 \end{cases}, (i = 1, 2, \dots) \text{ sind.}$$

Der @ Operator verkettet die Ziffern des linken Operanden mit den Ziffern des rechten Operanden in dieser Reihenfolge (die Ziffern entsprechen der Basis-10-Darstellung), und x ist eine natürliche Zahl ($1 \leq x \leq 99$). Wenn zum Beispiel $x = 3$ ist, enthält die Zeichenkette s die Werte 3, 4, 43, 434, 43443, Geben Sie für eine natürliche Zahl k ($1 \leq k \leq 30$) die Anzahl der Ziffern des Terms in der Zeichenkette s an, die der Vorgänger des k I-stelligen Term ist, wobei k I die kleinste Zahl mit der Eigenschaft ist, dass $k \leq kI \leq 30$ ist und es einen kI -stelligen Term gibt.

- A. wenn $x = 15$ und $k = 8$, ist die Anzahl der Ziffern des gesuchten Terms 6.
- B. wenn $x = 2$ und $k = 6$, ist die Anzahl der Ziffern des gesuchten Terms 6.
- C. wenn $x = 14$ und $k = 27$, ist die Anzahl der Ziffern des gesuchten Terms 26.
- D. wenn $x = 5$ und $k = 12$, ist die Anzahl der Ziffern des gesuchten Terms 8.

23. Wir betrachten den folgenden rekursiven $\text{fibonacci}(n)$ Algorithmus, wobei n eine natürliche Zahl ist ($1 \leq n \leq 100$). Ermitteln Sie, wie oft die Meldung "Aici" bei einem $\text{fibonacci}(n)$ -Aufruf angezeigt wird.

```

Algorithm fibonacci(n):
  If n ≤ 1 then
    Write "Aici"
    return 1
  else
    return fibonacci(n - 1) + fibonacci(n - 2)
  EndIf
EndAlgorithm

```

- A. fibonacci(n) mal.
- B. fibonacci(n-1) mal.
- C. fibonacci(n)-1 mal.
- D. fibonacci(n) - fibonacci(n-1) mal.

24. Betrachten Sie den Ausdruck: $E(x) = a_0 + a_1*x + a_2*x^2 + a_3*x^3 + a_4*x^4$, wobei a_0, a_1, a_2, a_3, a_4 und x reelle Zahlen ungleich Null sind. Die Mindestanzahl der Multiplikationen, die zur Berechnung des Wertes des Ausdrucks $E(x)$ erforderlich sind, beträgt:

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

25. Wir betrachten den Algorithmus $f(x, n)$, wobei x, n natürliche Zahlen sind und $x > 0$.

```

1. Algorithm f(x, n):
2.   If n = 0 then
3.     return 1
4.   EndIf
5.   m ← n DIV 2
6.   p ← f(x, m)
7.   If n MOD 2 = 0 then
8.     return p * p
9.   EndIf
10.  return x * p * p
11. EndAlgorithm

```

25a. Welche der folgenden Aussagen sind zutreffend?

- A. Der Algorithmus gibt x^n zurück, indem er ungefähr n rekursive Aufrufe macht.
- B. Der Algorithmus gibt x^n zurück, indem er ungefähr $\log_2 n$ rekursive Aufrufe durchführt.
- C. Der Algorithmus liefert x^n nur dann, wenn n eine Potenz von 2 ist
- D. Der Algorithmus liefert x^n nur dann, wenn n gerade ist.

25b. Falls wir die Zeile 10 durch:

```
10. return x * f(x, n - 1)
```

ersetzen, welche der folgenden Aussagen sind zutreffend?

- A. Der Algorithmus liefert nicht mehr x^n
- B. Der Algorithmus liefert x^{n+1}
- C. Der Algorithmus führt ungefähr n^2 rekursive Aufrufe durch.
- D. Der Algorithmus liefert x^n

26. Wir betrachten den Algorithmus $f_2(a, b)$ mit den Parametern a und b , die natürliche Zahlen sind, und den Algorithmus $f(arr, i, n, p)$, der als Parameter die Zeichenkette arr mit n ganzen Zahlen ($arr[1], arr[2], \dots, arr[n]$) und die ganzen Zahlen i und p hat.

```

Algorithm f2(a, b):
  If a > b then
    return a
  else
    return b
  EndIf
EndAlgorithm

Algorithm f(arr, i, n, p):
  If i = n then
    return 0
  EndIf
  n1 ← f(arr, i + 1, n, p)
  n2 ← 0
  If p + 1 ≠ i then
    n2 ← f(arr, i + 1, n, i) + arr[i]
  EndIf
  return f2(n1, n2)
EndAlgorithm

```

Geben Sie das Ergebnis des Aufrufs $f(arr, 1, 9, -10)$ an, wenn die Zeichenfolge **arr** die Werte (10, 1, 5, 4, 7, 12, 1, 12, 6) enthält.

- A. 24
- B. 37
- C. 39
- D. 56

27. Wir betrachten den Algorithmus $f(n)$, der eine natürliche Zahl n als Parameter hat und eine natürliche Zahl zurückgibt.

```
Algorithm f(n):
  j ← n
  While j > 1 execute
    i ← 1
    While i ≤ n4 execute
      i ← 4 * i
      Write "*"
    EndWhile
    If j DIV 2 > 1 then
      Write " "
    EndIf
    j ← j DIV 2
  EndWhile
  return j
EndAlgorithm
```

27a. In welche der folgenden Komplexitätsklassen fällt die Zeitkomplexität des Algorithmus?

- A. $O(\log_2 n)$
- B. $O(\log_2^2 n)$
- C. $O(\log_4^2 n)$
- D. $O(\log_2 \log_4 n)$

27b. Welche der folgenden Aussagen sind zutreffend?

- A. Wenn $n = 10$ ist, zeigt der Algorithmus Gruppen von jeweils 7 Sternen an, die durch ein Leerzeichen getrennt sind.
- B. Wenn $n = 20$ ist, zeigt der Algorithmus 4 Gruppen von Sternen und 4 Leerzeichen an.
- C. Wenn $n = 25$ ist, zeigt der Algorithmus 48 Sterne an, und nach jeder Gruppe wird ein Leerzeichen angezeigt.
- D. Wenn $n = 100$ ist, zeigt der Algorithmus 84 Sterne und 5 Leerzeichen an.

Wettbewerb Mathe-Info 25 März 2022

Schriftliche Prüfung in Informatik

PUNKTEANZAHL & LÖSUNGEN

23 – 03 - 2022

ANFANGSPUNKTEANZAHL: 10 punkte

1	C	3 punkte
2	ABD	3 punkte
3	AD	3 punkte
4	C	3 punkte
5	B	3 punkte
6	D	3 punkte
7	ABD	3 punkte
8	C	3 punkte
9	B	3 punkte
10	C	3 punkte
11	C	3 punkte
12	ACD	3 punkte
13	C	3 punkte
14	B	3 punkte
15	BD	3 punkte
16	B	3 punkte
17	CD	3 punkte
18	AC	3 punkte
19	BD	3 punkte
21	AC	3 punkte
22	AD	3 punkte
23	A	3 punkte
24	A	3 punkte
26	C	3 punkte
20a	B	3 punkte
20b	D	3 punkte
25a	B	3 punkte
25b	D	3 punkte
27a	BC	3 punkte
27b	AD	3 punkte