

Questa è la Prima fase delle Olimpiadi Svizzere d'Informatica. Il test consiste in 16 domande a scelta multipla, ciascuna del valore di 1 punto.

Regole: Devi completare l'esame da solo e senza l'aiuto dell'intelligenza artificiale. L'uso della calcolatrice è consentito, ma non necessario.

Domande generali (10 domande): Contiene domande che implicano il ragionamento logico e semplici concetti informatici.

Domande di programmazione (6 domande): Contiene domande con pseudocodice di facile comprensione e alcune domande a riguardo. Non è richiesta una precedente conoscenza di programmazione.

Importante: I problemi in ogni sezione sono ordinati per difficoltà. Ricordati di provare a risolvere problemi di entrambe le sezioni.

Limite di tempo: 40 minuti

Durata: 40 minutes

1 Domande generali

Queste sono le domande generali. Le seguenti 10 domande sono ordinate per difficoltà. Tutte le domande valgono 1 punto. Non è prevista alcuna penalità per le risposte errate.

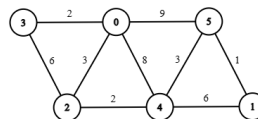
Domanda 2

Data la sequenza $[5, 2, 1, 3]$, qual è il numero minimo di passaggi necessari per ordinare la sequenza in ordine crescente? In ogni passaggio, è possibile scambiare due elementi adiacenti. Per esempio, $[5, 2, 1, 3]$ può diventare $[5, 2, 3, 1]$ in un passaggio, ma diventare $[1, 2, 5, 3]$ in un passaggio non è possibile.

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

Domanda 3

Ci sono 6 città (numerate 0, 1, 2, 3, 4, 5) collegate da strade a doppio senso. Ogni strada collega due città e richiede un certo numero di minuti per essere percorsa, come indicato nel diagramma sopra. Qual è il tempo minimo necessario per viaggiare dalla città 3 alla città 1?



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

Domanda 4

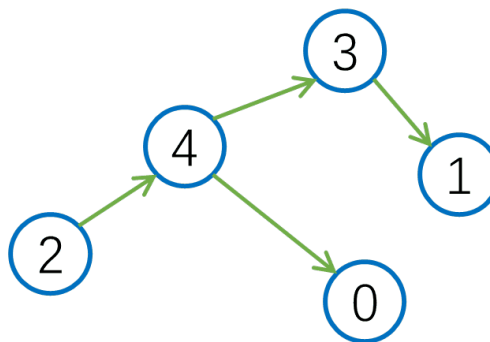
Hai 9 paperelle gialle e 99 paperelle bianche. Puoi comprare una paperella gialla in più per 1 CHF e una paperella bianca per 1 CHF. Puoi dipingere qualsiasi paperella di rosso gratuitamente. Qual è il numero minimo di CHF che devi pagare per avere lo stesso numero di paperelle di ogni colore, dato che non puoi sbarazzarti delle paperelle?

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

Domanda 5

Ci sono 5 città (numerate 0, 1, 2, 3, 4) collegate da strade a senso unico. La mappa sopra mostra le strade attuali tra le città. Non è possibile partire da una città e visitare ogni altra città esattamente una volta usando le strade mostrate.

Quale delle seguenti nuove strade, se aggiunta alla mappa, renderebbe possibile visitare tutte le città esattamente una volta, partendo da una città e seguendo la direzione delle strade?



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

Domanda 6

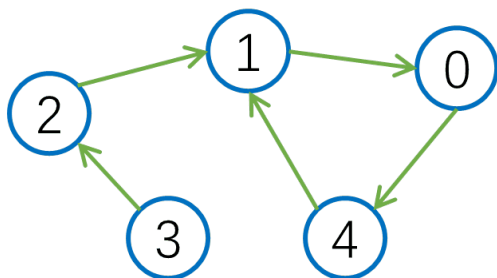
Il topo Stoffl ha pensato a un numero tra 1 e 25 e non te lo dice. Puoi provare a indovinare una sequenza contigua di numeri (per esempio 4 – 7), e il topo Stoffl ti dirà se il suo numero è nel tuo intervallo, al di sopra o al di sotto. Qual è il numero minimo di tentativi necessari per essere sicuri di trovare il numero di Stoffl?

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

Domanda 7

Ci sono 5 città (numerate 0, 1, 2, 3, 4) collegate da strade a senso unico. È permesso aggiungere una nuova strada a senso unico tra due città tra cui non esista già una strada (in nessuna delle due direzioni).

In quanti modi diversi è possibile aggiungere tale strada in modo che, dopo averla aggiunta, diventi possibile viaggiare da qualsiasi città a qualsiasi altra città seguendo la direzione delle strade?



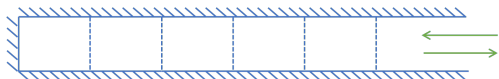
- A. 0 B. 1 C. 2 D. 3 E. 4

Domanda 8

C'è un parcheggio stretto con 5 posti auto in linea retta. In particolare, un'auto può uscire solo se non ci sono altre auto parcheggiate tra essa e l'uscita. Cinque auto — A , B , C , D , ed E — entrano nel parcheggio in quest'ordine (A per prima, poi B , e così via).

Nota: Le auto possono uscire prima che tutte e cinque le auto siano entrate.

Quale dei seguenti ordini di uscita non è possibile?



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

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

Domanda 9

Ti viene fornita una lista di 4 numeri distinti. Puoi solo confrontare due numeri alla volta per determinare quale sia il più grande. Indipendentemente dall'ordine iniziale dei numeri, qual è il numero minimo di confronti richiesto per garantire che i numeri possano essere completamente ordinati in ordine crescente?

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

Domanda 10

La nazione di Crazystoffl è composta da 2026 isole numerate da 0 a 2025. È noto che ogni isola x è collegata da un volo di andata e ritorno con l'isola $\lfloor \frac{x}{2} \rfloor$. Qual è il numero minimo di voli che si devono prendere per andare dall'isola numero 125 all'isola 1025?

Commento: $\lfloor y \rfloor$ è il più grande intero che non supera il valore di y . Per esempio $\lfloor 3.14 \rfloor = 3$, $\lfloor 7 \rfloor = 7$, $\lfloor \frac{11}{2} \rfloor = 5$.

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

Domanda 11

In quanti modi è possibile ricoprire interamente una griglia 4×14 con delle piastrelle 1×4 in modo tale che nessuna piastrella si sovrapponga? Nota che le piastrelle possono essere ruotate.

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

2 Domande sul programma

Queste sono le domande del programma. Le seguenti 6 domande sono ordinate in base alla difficoltà. Tutte le domande valgono 1 punto. Non sono previste penalità per le risposte errate.

Domanda 12

Nel seguente programma, qual è l'output del programma?

Note

- **Set** assegna un valore a un nome di variabile specifico.

- **If**, **Else If**, o **Else** sono chiamati condizionali. I condizionali eseguono un certo blocco di codice se la condizione (specificata sulla stessa riga) è vera. L'indentazione mostra quali righe di codice appartengono a ciascun condizionale.

- **Print** stampa il valore di una variabile.

```

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

Domanda 13

Nel seguente programma, qual è l'output del programma?

Note

-Un ciclo **For** ripete il suo codice un determinato numero di volte, come indicato dall'intervallo inclusivo. Per esempio, **For x from 2 to 4** itererebbe il codice tre volte da 2 a 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

Domanda 14

Nel seguente programma, qual è l'output del programma?

Note

-Un ciclo **While** continuerà a essere eseguito **finché la sua condizione rimane vera**. Il ciclo controlla la sua condizione **all'inizio** di ogni iterazione. Una volta che la condizione diventa falsa, il ciclo si ferma e il programma continua con l'istruzione successiva al ciclo.

- $x == y$ sarà considerato vero se x e y sono uguali, e falso in ogni altro caso. Per esempio $1 == 2$ è falso e $5 == 5$ è vero.

```

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

Domanda 15

Nel seguente programma, qual è l'output del programma?

Note

-Una **Function** (Funzione) definisce un insieme di codice che può essere richiamato più volte. Di solito contiene un input e un valore di ritorno.

- $a \bmod b$ significa il resto della divisione di a per b , es. $7 \bmod 3 = 1$ e $8 \bmod 4 = 0$.

```

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. $2^{248} + 6$ D. $2^{49} + 2^{50} + 2^{51}$
E. Nessuna delle precedenti

Domanda 16

Nel seguente programma, qual è l'output del programma?

Note

- Gli **Indices** sono numeri di posizione che indicano la posizione di qualcosa in una lista. Sono indicizzati da 0. Per esempio, se lista = (2, 4, 6) allora lista[0] = 2 e lista[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

Domanda 17

Quanti valori diversi può assumere inizialmente K (campo giallo) affinché il nostro programma restituisca “Yay!” per un N scelto liberamente (campo giallo)?

Commento: N e K devono essere numeri interi.

```
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. Infiniti valori possibili