# OliCyber.IT 2026 - Selezione Scolastica
# Soluzioni commentate

## Contenuti

# 1 Domanda 1

## 1.1 Domanda

Ci sono tre persone, Francesco, Lorenzo e Matteo. Uno di loro è un white-hat, uno un black-hat e uno un grey-hat.

I white-hat dicono sempre la verità, i black-hat mentono sempre e i grey-hat possono sia mentire sia dire la verità.

- Francesco dice: "Matteo è un black-hat".
- Lorenzo dice: "Francesco è un white-hat".
- Matteo dice: "Io sono un grey-hat".

Chi è il white-hat, chi il black-hat e chi il grey-hat?

## 1.2 Risposte

(A) `Francesco white-hat, Lorenzo grey-hat, Matteo black-hat`

(B) `Francesco grey-hat, Lorenzo black-hat, Matteo white-hat`

(C) `Francesco black-hat, Lorenzo grey-hat, Matteo white-hat`

(D) `Francesco grey-hat, Lorenzo white-hat, Matteo black-hat`

## 1.3 Soluzione proposta

La risposta corretta è `(A) Francesco white-hat, Lorenzo grey-hat, Matteo black-hat`.

Sapendo che c'è un solo white-hat, notiamo subito che non può essere Lorenzo, altrimenti, dicendo la verità, dovrebbe esserlo anche Francesco.

Vediamo che neanche Matteo può essere il white-hat, in quanto, se fosse così, starebbe mentendo dicendo di essere un grey-hat.

Quindi il white-hat deve essere Francesco. Dunque, dato che dice la verità, Matteo è un black-hat (e ciò è coerente con il fatto che sta mentendo dicendo di essere un grey-hat). Mancando solo il grey-hat, questi deve essere Lorenzo, che sta dicendo la verità.

La risposta corretta è dunque la `(A)`.

# 2 Domanda 2

## 2.1 Domanda

A una conferenza di cybersecurity ci sono 2025 persone, che possono essere white-hat o black-hat. I white-hat dicono sempre la verità, mentre i black-hat mentono sempre. Puoi fare loro solo domande del tipo "persona A, che tipo di cappello è la persona B?", dove A e B sono una qualsiasi coppia di persone, alla quale la persona A risponderà indicando il colore. Sai anche che i white-hat sono in numero dispari. Quante domande devi fare come minimo per trovare con certezza almeno un white-hat?

## 2.2 Risposte

(A) `1012`

(B) `1013`

(C) `2024`

(D) `2025`

## 2.3 Soluzione proposta

La risposta corretta è `(B) 1013`.

Osserviamo, per prima cosa, che l'unica informazione che otteniamo dalla domanda che possiamo fare è se A e B sono dello stesso tipo oppure no, ma non otteniamo nulla su chi è cosa. Infatti se A risponde che B è un white-hat, allora abbiamo che A e B possono essere o entrambi white-hat o entrambi black-hat. Invece, se A risponde che B è un black-hat, abbiamo che o A è un white-hat e B un black-hat o viceversa.

Dimostriamo ora che possiamo trovare con certezza un white-hat facendo 1013 domande. Per farlo dividiamo le 2025 persone in 1012 coppie, con una persona X che rimane fuori. Ad ogni coppia, chiamando A e B le persone della coppia, chiediamo ad A di B, scoprendo così se sono dello stesso tipo oppure no.

Notiamo che, se una coppia è dello stesso tipo, sia che siano 2 white-hat che 2 black-hat, ciò non cambia la parità dei white-hat rimasti. Se invece la coppia è di tipo diverso, allora conterrà un white-hat ed un black-hat, cambiando la parità dei white-hat rimasti.

Quindi, dopo aver fatto le 1012 domande alle coppie, avremo due casi. Se abbiamo ottenuto un numero pari di coppie di tipo diverso, allora la persona X rimasta deve essere un white-hat e in questo caso abbiamo finito.
Se invece abbiamo ottenuto un numero dispari di coppie di tipo diverso, allora la persona X rimasta deve essere un black-hat. Prendiamo quindi una delle coppie di tipo diverso (che sappiamo esistere perché sono in numero dispari) e chiamiamo A e B le due persone. A questo punto basta chiedere ad X che tipo di cappello è A. Se risponde white-hat, dato che mente, B sarà il white-hat, altrimenti lo è A. In entrambi i casi abbiamo trovato un white-hat.

Dimostriamo ora che, con 1012 domande, esisterà sempre un caso in cui non riusciamo a trovare con certezza un white-hat. Consideriamo le 2025 persone come nodi di un grafo e, ogni volta che chiediamo ad una persona A di una persona B, colleghiamo un arco tra A e B. Dopo le 1012 domande il grafo sarà diviso in alcune componenti connesse al loro interno e separate dalle altre.

Notiamo che, in una componente connessa, quando sappiamo il tipo di una persona, lo conosciamo anche di tutte le altre, seguendo le implicazioni date dagli archi (che possono essere "stesso tipo" o "diverso tipo"). Quindi, per una componente connessa, ci saranno due configurazioni possibili, cioè la configurazione X e la configurazione $\overline{\text{X}}$ in cui invertiamo il tipo di ogni persona.

Consideriamo una componente connessa con un numero pari di nodi. Allora il numero di white-hat in X e in $\overline{\text{X}}$ avrà la stessa parità. Quindi, dalla condizione che il numero di white-hat è dispari, non potremo mai distinguere tra queste due, ovvero non potremo mai trovare un white-hat con certezza all'interno di una componente pari.

Consideriamo ora le componenti connesse con un numero dispari di nodi e dividiamo in due casi. Se c'è una sola componente connessa di dimensione dispari, allora siamo nel caso già descritto sopra in cui abbiamo 1012

coppie e una persona rimasta fuori. Allora abbiamo già visto che in questo caso è possibile che la persona rimasta fuori sia un black-hat e non abbiamo un'altra domanda a disposizione per trovare un white-hat.
Se invece c'è più di una componente dispari osserviamo prima la seguente cosa: per le componenti dispari, il numero di white-hat in X e in $\overline{\text{X}}$ cambia di parità. Però, se abbiamo due componenti dispari $C_1$ e $C_2$, con possibili configurazioni $(X_1, \overline{X_1})$ e $(X_2, \overline{X_2})$, rispettivamente, allora il numero di white-hat totali nelle configurazioni $(X_1, X_2)$ e $(\overline{X_1}, \overline{X_2})$ ha la stessa parità. Questo vuol dire che, se abbiamo più di una componente dispari, non potremo mai identificare un white-hat all'interno di una di queste, perché invertendo la sua configurazione e quella di un'altra componente dispari otteniamo un'altra soluzione valida che soddisfa i requisiti di parità.

In questo modo abbiamo dimostrato che, con 1012 domande, non potremo mai identificare un white-hat né all'interno di una componente connessa pari né all'interno di una dispari, ovvero 1013 è il numero minimo di domande necessarie.

# 3 Domanda 3

## 3.1 Domanda

Considera la seguente funzione:

```
function encrypt(m, len):
  for i = 0,1,...,len-1:
    m[i] = chr((ord(m[i]) * ord(m[(i+2)%len])) % 26)
  return m
```

dove `ord` è la funzione che mappa A a 0, B a 1 e così via, mentre `chr` è la sua funzione inversa. Quante possibili stringhe `m` composte solo da lettere maiuscole (anche senza senso) ci sono tali che `encrypt(m, 6) = "ABABAB"`?

## 3.2 Risposte

(A) `0`

(B) `1`

(C) `296`

(D) `905`

## 3.3 Soluzione proposta

La risposta corretta è `(D) 905`.

Notiamo che la funzione sovrascrive la lista `m`. Chiamiamo $m_0, m_1, m_2, m_3, m_4, m_5$ i numeri corrispondenti ai caratteri del messaggio `m`. Dato che la funzione modifica la lista "in place", abbiamo che, all'iterazione `i = 5`, l'ultimo carattere deve essere stato sotituito con

`m[5] = chr((ord(m[5]) * ord(m[1])) % 26)`.

Ma, essendo `m[1]` già stato sostituito con `B` e, sapendo che il risultato finale di `m[5]` è anch'esso una `B`, abbiamo l'equazione

$$1 = m_5 \cdot 1 \pmod{26}$$

ovvero $m_5 = 1$, cioè l'ultimo carattere del messaggio `m` deve essere una `B`. Andando a ritroso possiamo fare il ragionamento analogo per $m_3$ ed $m_1$, ottenendo che anch'essi, nel messaggio originale, devono essere delle `B`. Rimane quindi da determinare i possibili valori di $m_0, m_2, m_4$.

Con un ragionamento analogo a quello sopra, otteniamo l'equazione per $m_4$:

$$0 = m_4 \cdot 0 \pmod{26}$$

che però non aggiunge restrizioni ad $m_4$ perché sempre vera. Per $m_2$ e $m_0$ abbiamo invece le equazioni

$$\begin{cases} m_2 \cdot m_4 = 0 \pmod{26} \\ m_0 \cdot m_2 = 0 \pmod{26} \end{cases}$$

Per vedere quante soluzioni possibili ci sono di questo sistema, dividiamo nei vari casi per $m_2$.

- $m_2 = 0$: `un caso` $\rightarrow$ allora $m_0$ e $m_4$ possono avere qualsiasi valore valore tra 0 e 25, dando $26^2 = 676$ valori possibili.
- $m_2$ `pari ma diverso da 0:  dodici casi` $\rightarrow$ allora $m_0$ e $m_4$ devono essere multipli di 13, avendo quindi 2 possibilità ciascuno (0 e 13) e dando $12 \cdot 2^2 = 48$ valori possibili.
- $m_2 = 13$: `un caso` $\rightarrow$ allora $m_0$ e $m_4$ devono per forza essere pari, avendo quindi 13 possibilità ciascuno e dando $13^2 = 169$ valori possibili.

- `tutti gli altri casi:` `dodici casi` → in questi casi sia $m_0$ che $m_4$ devono essere 0, dando $12 \cdot 1^2 = 12$ valori possibili.

In totale, per $(m_0, m_2, m_4)$ abbiamo quindi $676 + 48 + 169 + 12 = 905$ soluzioni. Dato che per le altre posizioni c'è una sola possibilità, le possibili stringhe `m` tali che `encrypt(m, 6) = "ABABAB"` sono 905.

# 4 Domanda 4

## 4.1 Domanda

Considera la seguente funzione:

```
function f(l,k):
  for i = 0,...,k-1:
    if l[0] % 2 == 0:
        l = concat(l, l)
    l = rotl(l, 1)
  return len(l)
```

dove `concat(a, b)` è la funzione che concatena le due liste `a` e `b`, `len(l)` è la funzione che calcola la lunghezza della lista `l` e `rotl(l, k)` ruota la lista a sinistra di `k` posizioni.

Per esempio:

- `concat([1, 2, 3], [4, 5]) = [1, 2, 3, 4, 5]`
- `len([3, 4, 5]) = 3`
- `rotl([5, 6, 7, 8, 9], 1) = [6, 7, 8, 9, 5]`

Qual è il risultato di `f([1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0], 1000)`?

## 4.2 Risposte

(A) $11 \cdot 2^{998}$

(B) $11 \cdot 2^{455}$

(C) $11 \cdot 2^{999}$

(D) $11 \cdot 2^{454}$

## 4.3 Soluzione proposta

La risposta corretta è `(D)` $11 \cdot 2^{454}$.

Osserviamo, per prima cosa, che, dato che l'operazione `concat` è applicata soltanto due copie della stessa lista, le operazioni `concat` e `rotl` commutano, ovvero se facciamo prima una e poi l'altra o viceversa otteniamo lo stesso risultato. Infatti, se consideriamo una lista generica del tipo `l = [`$a_1$`, ..., `$a_n$`]` osserviamo cosa accade facendo prima l'operazione di `concat` e poi quella di `rotl`:

$$\texttt{l = concat(l, l)} = [a_1, \ldots, a_n, a_1, \ldots, a_n]$$
$$\texttt{rotl(l, 1)} = [a_2, \ldots, a_n, a_1, a_2, \ldots, a_n, a_1]$$

Eseguendo invece prima l'operazione di `rotl` e poi quella di `concat` abbiamo:

$$\texttt{l = rotl(l, 1)} = [a_2, \ldots, a_n, a_1]$$
$$\texttt{concat(l, l)} = [a_2, \ldots, a_n, a_1, a_2, \ldots, a_n, a_1]$$

ottenendo quindi lo stesso risultato.

Questo vuol dire, che se consideriamo le prime 11 iterazioni, possiamo prima effettuare tutte le operazioni di `rotl` e poi tutte quelle di `concat`. Tuttavia ruotando la lista per 11 volte di una posizione essa ritorna uguale a sé stessa, quindi le prime non hanno nessun effetto. Le operazioni di `concat` effettuate sono invece 5, perché la lista contiene al suo interno 5 zeri.

Quindi ogni 11 iterazioni la lista verrà raddoppiata 5 volte. Dato che $1000 = 90 \cdot 11 + 10$ verranno effettuati 90 cicli da 11, all'interno dei quali la lista raddoppia per 5 volte, raggiungendo quindi una lunghezza di

$11 \cdot 2^{90 \cdot 5} = 11 \cdot 2^{450}$.

Alla fine di questi cicli, la lista che abbiamo è esattamente quella di partenza ripetuta per $2^{450}$ volte. Quindi, nelle 10 iterazioni rimanenti, verranno considerati i primi 10 elementi della lista, all'interno dei quali ci sono solamente 4 zeri. La lista viene quindi raddoppiata altre 4 volte, raggiungendo una lunghezza totale di $11 \cdot 2^{454}$.

# 5 Domanda 5

## 5.1 Domanda

Considera la seguente funzione:

```
function f(a, b):
    res, carry, ab2, i = 0, 0, 0, 0
    while a or b:
        axb = (a&1) ^ (b&1)
        res |= (axb ^ ab2 ^ carry) << i
        carry = (axb & ab2) | (axb & carry)
        ab2 = (a&1) & (b&1)
        a >>= 1
        b >>= 1
        i += 1
    res |= (ab2 ^ carry) << i
    return res
```

dove `&`, `|`, `^`, `<<`, `>>` rappresentano rispettivamente le operazioni di bitwise-and, bitwise-or, bitwise-xor, bitwise-shift a sinistra e bitwise-shift a destra.

Cosa calcola questa funzione?

## 5.2 Risposte

(A) `a + b`

(B) `a * b`

(C) $a^b$

(D) `a / b (divisione intera)`

## 5.3 Soluzione proposta

La risposta corretta è `(A) a + b`.

La funzione implementa l'operazione di addizione sfruttando la proprietà `a + b = a^b + 2·(a&b)`.
La somma tra `a^b` e `2·(a&b)` viene effettuata bit per bit all'interno del ciclo `while`.

- Nella variabile `axb` viene salvato il bit attuale di `a^b`.
- Nella variabile `carry` viene salvato il riporto sulla somma dei due bit attuali.
- Nella variabile `ab2` viene salvato il bit corrispondente di `2·(a&b)`.

Ad ogni iterazione viene aggiunto il bit attuale al risultato `res` e calcolato il nuovo riporto e il nuovo valore di `ab2` per l'iterazione successiva.

Infine viene aggiunto l'ultimo bit eventuale derivante dal riporto o da `ab2`.

# 6 Domanda 6

## 6.1 Domanda

Ti trovi davanti ad un cifrario sconosciuto che prende in input un messaggio e ci applica la seguente funzione:

```
function encrypt(m, k, len_m, len_k):
  for i = 0,1,...,len_m-1:
    m[i] = chr((ord(m[i]) + k[i % len_k]) % 26)
  return m
```

dove `ord` è la funzione che mappa A a 0, B a 1 e così via, mentre `chr` è la sua funzione inversa. Quale può essere un possibile messaggio originale `m` del messaggio cifrato `encrypt(m, k, 20, 4) = VKTUQRJDAHFSUZVEGTUJ` per un qualche valore di `k`?

## 6.2 Risposte

(A) `GARAOLICYBERCROCETTE`

(B) `TESTOLICYBERSTUDENTI`

(C) `GABIBBOCYBERSECURITY`

(D) `SUPERGABIBBOOLICYBER`

## 6.3 Soluzione proposta

La risposta corretta è `(B) TESTOLICYBERSTUDENTI`.

Il cifrario implementato è il cifrario classico di Vigenère, con una chiave lunga 4 caratteri. Dunque, sui caratteri del testo cifrato le cui posizioni distano un multiplo di 4, il cifrario si comporta come un cifrario di Cesare. Dividendo il testo cifrato in blocchi da 4 abbiamo:

```
VKTU QRJD AHFS UZVE GTUJ
```

Notiamo che la prima lettera del primo blocco e la prima lettera del quarto blocco, che sono `V` ed `U`, hanno distanza 1. Essendo state cifrate con lo stesso carattere della chiave, anche nel testo in chiaro le lettere corrispondenti devono avere distanza 1. Dividiamo quindi anche le possibili soluzioni in blocchi da 4:

```
GARA OLIC YBER CROC ETTE
TEST OLIC YBER STUD ENTI
GABI BBOC YBER SECU RITY
SUPE RGAB IBBO OLIC YBER
```

Notiamo che l'unica risposta che soddisfa la condizione menzionata precedentemente è la seconda, `TESTOLICYBERSTUDENTI`, la cui prima lettera del primo blocco è una `T` e la prima lettera del quarto blocco è una `S`, che hanno distanza 1.

# 7 Domanda 7

## 7.1 Domanda

Ad una conferenza di cybersecurity ci sono white-hat e black-hat. I white-hat dicono sempre la verità, mentre i black-hat mentono sempre. Ciascuno, inoltre, può essere un webber o un pwner.

Fra un talk e l'altro, incontri Abibbo e Babibbo, che ti dicono:

- Abibbo: Babibbo è un black-hat e un webber.
- Babibbo: Abibbo è un white-hat.
- Abibbo: Io sono un pwner.

Cosa sono Abibbo e Babibbo?

## 7.2 Risposte

(A) `Abibbo webber, Babibbo pwner, entrambi black-hat`

(B) `Abibbo pwner, Babibbo webber, entrambi black-hat`

(C) `Abibbo white-hat webber, Babibbo black-hat pwner`

(D) `Abibbo white-hat pwner, Babibbo black-hat webber`

## 7.3 Soluzione proposta

La risposta corretta è `(A) Abibbo webber, Babibbo pwner, entrambi black-hat`.

Supponiamo per prima cosa che Babibbo sia un white-hat. Allora dovrebbe esserlo anche Abibbbo, che però afferma che Babibbo sia un black-hat, il che ci porta ad una contraddizione.

Dunque Babibbo deve essere un black-hat e, dato che sta mentendo, deve esserlo anche Abibbo. Considerando la prima affermazione di Abibbo, questa è costituita da due proposizioni legate da un `and` logico, ovvero `Babibbo è un black-hat` e `Babibbo è un webber`.

Dato che Abibbo sta mentendo, almeno una di queste affermazioni deve essere falsa. Essendo la prima vera, quella falsa deve per forza essere la seconda. Quindi Babibbo non è un webber ma un pwner.

Infine, dall'ultima affermazione di Abibbo, che anch'essa deve essere falsa, deduciamo che Abibbo è un webber.

La risposta corretta è quindi la `(A)`

# 8 Domanda 8

## 8.1 Domanda

Considera la seguente equazione:

`(X ^ 0xd4bc2816c72cd004) | (Y ^ 0xe8c152e48c67ec32) =` $2^{64}$ `- 1`

Quante soluzioni esistono con $X$ e $Y$ tra 0 e $2^{128} - 1$?

*Nota: l'OR bitwise, indicato da |, è l'operazione bit a bit definita dalla seguente tabella di verità:* $0 \mid 0 = 0$, $1 \mid 0 = 1$, $0 \mid 1 = 1$, $1 \mid 1 = 1$.

*Lo XOR bitwise, indicato da ^, è l'operazione bit a bit definita dalla seguente tabella di verità:* $0 \hat{} 0 = 0$, $1 \hat{} 0 = 1$, $0 \hat{} 1 = 1$, $1 \hat{} 1 = 0$.

## 8.2 Risposte

(A) $3^{64} \cdot 2^{64}$

(B) $3^{64} \cdot 2^{128}$

(C) $3^{64}$

(D) $4^{128}$

## 8.3 Soluzione proposta

La risposta corretta è `(C)` $3^{64}$.

Osserviamo per prima cosa che le operazioni di XOR all'interno delle parentesi non influiscono sul numero di soluzioni. Infatti, se consideriamo `Z,W` soluzioni dell'equazione `Z | W =` $2^{64}$ `- 1`,
allora `X = Z ^ 0xd4bc2816c72cd004` e `Y = W ^ 0xe8c152e48c67ec32` saranno soluzioni dell'equazione del problema. Viceversa, se `X` e `Y` sono soluzioni dell'equazione del problema, allora `Z = X ^ 0xd4bc2816c72cd004` e `W = Y ^ 0xe8c152e48c67ec32` sono soluzioni dell'equazione `Z | W =` $2^{64}$ `- 1`.

Per contare il numero di soluzioni di questa equazione, con `Z` e `W` compresi tra 0 e $2^{128} - 1$, notiamo che $2^{64} - 1$ è composto da 64 bit tutti ad 1. Per ognuno di questi bit, abbiamo 3 possibili scelte per i bit nella posizione corrispondente di `Z` e `W`, ovvero $(1, 0), (0, 1)$ e $(1, 1)$.

I 64 bit più significativi di `Z` e `W` dovranno invece essere tutti 0, altrimenti `Z | W` sarebbe maggiore o uguale di $2^{64}$.

In totale abbiamo quindi $3^{64}$ possibili soluzioni.

# 9 Domanda 9

## 9.1 Domanda

Stai tentando di sbloccare una serratura elettronica recuperata da una cassaforte. La serratura ha un indicatore rotante con 3 posizioni: 0 (Bloccato), 1 (Attesa), 2 (Sblocco). L'indicatore parte dalla posizione 0.

Il firmware accetta una stringa di comando lunga esattamente 128 bit, composta da due tipi di istruzioni:

- FORWARD: ruota l'indicatore in avanti (senso orario). Questa istruzione è complessa e consuma 7 bit.
- BACKWARD: ruota l'indicatore indietro (senso antiorario). Questa istruzione è semplice e consuma 1 bit.

Per sbloccare la serratura, l'indicatore deve fermarsi esattamente sulla posizione 2. Inoltre il numero di istruzioni FORWARD deve essere maggiore del numero di BACKWARD.

Quante istruzioni FORWARD e quante BACKWARD devi inviare?

*Esempio: se la stringa di comando fosse lunga 30 bit, una soluzione possibile sarebbe da 4 FORWARD e 2 BACKWARD, perché così la serratura farebbe 4 salti avanti e 2 indietro, fermandosi sul 2.*

## 9.2 Risposte

(A) `15 FORWARD e 13 BACKWARD`

(B) `16 FORWARD e 14 BACKWARD`

(C) `17 FORWARD e 9 BACKWARD`

(D) `18 FORWARD e 2 BACKWARD`

## 9.3 Soluzione proposta

La risposta corretta è `(C) 17 FORWARD e 9 BACKWARD`.

Chiamiamo $x$ il numero di istruzioni FORWARD e $y$ il numero di istruzioni BACKWARD. Dato che la lunghezza della stringa di comando deve essere esattamente 128 bit e le istruzioni FORWARD occupano 7 bit, mentre quelle BACKWARD solo 1, otteniamo che $x$ e $y$ devono soddisfare l'equazione

$$7x + y = 128.$$

Sapendo che le istruzioni FORWARD devono essere più di quelle BARCKWARD, abbiamo che $x > y$. Infine, dato che la serratura deve fermarsi esattamente sulla posizione 2, in totale dovrà aver fatto alcuni giri completi più due rotazioni. Abbiamo quindi l'equazione

$$x - y = 3k + 2$$

per un qualche $k \geq 0$ (notiamo che in questa equazione è già implicito il fatto che $x > y$. Nel complesso abbiamo quindi il sistema

$$\begin{cases} 7x + y = 128 \\ x - y = 3k + 2 \end{cases}$$

Sostituendo $x = y + 3k + 2$ nella prima equazione abbiamo

$$\begin{aligned} 8y + 21k + 14 &= 128 \\ \Rightarrow 8y &= 114 - 21k \end{aligned}$$

Testando per i possibili valori di $k$ abbiamo

- $(k = 0)$, $8y = 114$ non ha soluzione intera per $y$.

- $(k = 1)$, $8y = 93$ non ha soluzione intera per $y$.
- $(k = 2)$, $8y = 72$ quindi $y = 9$ e $x = 17$.
- $(k = 3)$, $8y = 51$ non ha soluzione intera per $y$.
- $(k = 4)$, $8y = 30$ non ha soluzione intera per $y$.
- $(k = 5)$, $8y = 9$ non ha soluzione intera per $y$.
- $(k \geq 6)$, il termine di destra è negativo, quindi non ci possono essere soluzioni positive per $y$.

L'unica soluzione è dunque $x = 17$ e $y = 9$ ovvero la `(C)`.

# 10 Domanda 10

## 10.1 Domanda

Di seguito trovi 3 affermazioni e 3 conclusioni. Dando le affermazioni per vere, quali conclusioni puoi dedurre con sicurezza?

Affermazioni:

1. Tutte le challenge di web sono challenge di misc.
2. Nessuna challenge di misc è una challenge di crypto.
3. Alcune challenge di pwn sono challenge di crypto.

Conclusioni:

1. Alcune challenge di web sono challenge di crypto.
2. Alcune challenge di pwn sono challenge di web.
3. Nessuna challenge di web è una challenge di crypto.

## 10.2 Risposte

(A) `Solo la conclusione 1.`

(B) `Solo la conclusione 2.`

(C) `Solo la conclusione 3.`

(D) `Almeno due conclusioni.`

## 10.3 Soluzione proposta

La risposta corretta è `(C) Solo la conclusione 3`.

Dato che le challenge di web sono un sottoinsieme di quelle di misc, mentre quelle di misc e quelle di crypto sono due insiemi disgiunti, possiamo dedurre che nessuna challenge di web sia una challenge di crypto. Quindi sicuramente possiamo dedurre la conclusione 3.

La conclusione 1 è falsa, perchè le challenge di web sono un sottoinsieme di quelle di misc, ma nessuna challenge di misc è una challenge di crypto.

Sulla conlcusione 2, invece, non possiamo dedurre nulla, in quanto l'unica informazione che sappiamo sulle challenge di pwn è che hanno intersezione non vuota con le challenge di crypto, ma non sappiamo nulla sulla loro intersezione con le challenge di web o di misc.

La risposta corretta è quindi la `(C)`

# 11 Domanda 11

## 11.1 Domanda

Gaspa-Rosso, Gaspa-Blu, Gaspa-Bianco e Gaspa-Verde si trovano ad un tavolo. A ognuno viene data una maglietta di colore diverso da indossare, tra rosso, blu, bianco e verde.

Gaspa-Blu dice "Hey, avete notato che stiamo tutti indossando una maglietta di colore diverso dal nostro nome?".

Il Gaspa che indossa una maglietta bianca risponde "Wow, hai ragione Gaspa-Blu!".

Il Gaspa con la maglietta verde poi dice "Menomale, il verde mi piace di più del colore del mio nome!".

Gaspa-Rosso, l'unico a non aver ancora parlato fin'ora, risponde "A me invece piaceva più il rosso!".

Tutte le frasi pronunciate sono vere. Riesci a capire chi sta indossando quale maglia?

## 11.2 Risposte

(A) `Gaspa-Rosso maglia blu, Gaspa-Blu maglia bianca, Gaspa-Bianco maglia verde, Gaspa-Verde maglia rossa`

(B) `Gaspa-Rosso maglia bianca, Gaspa-Blu maglia rossa, Gaspa-Bianco maglia verde, Gaspa-Verde maglia blu`

(C) `Gaspa-Rosso maglia blu, Gaspa-Blu maglia rossa, Gaspa-Bianco maglia verde, Gaspa-Verde maglia bianca`

(D) `Gaspa-Rosso maglia bianca, Gaspa-Blu maglia verde, Gaspa-Bianco maglia rossa, Gaspa-Verde maglia blu`

## 11.3 Soluzione proposta

La risposta corretta è `(C) Gaspa-Rosso maglia blu, Gaspa-Blu maglia rossa, Gaspa-Bianco maglia verde, Gaspa-Verde maglia bianca`.

Dall'affermazione di Gaspa-Blu sappiamo che egli non può indossare la maglietta blu. Inoltre, dal fatto che Gaspa-Rosso fosse **l'unico** a non aver ancora parlato, sappiamo che il Gaspa-Blu, il Gaspa con la maglietta bianca e il Gaspa con la maglietta verde devono essere 3 persone distinte.

Quindi Gaspa-Blu non può star indossando né la maglietta blu, né quella bianca, né quella verde, ovvero sta indossando quella rossa.

Analogamente, Gaspa-Rosso non può essere né il Gaspa con la maglietta bianca né quello con la maglietta verde, avendo parlato per ultimo. Non potendo indossare neanche la maglietta rossa, sta per forza indossando quella blu.

Rimangono quindi da associare solo Gaspa-Bianco e Gaspa-Verde alle magliette bianca e verde. Non potendo indossare la maglietta del colore del proprio nome, l'unica possibilità è che Gaspa-Bianco stia indossando quella verde e Gaspa-Verde quella bianca.

La risposta corretta è quindi la `(C)`.

# 12 Domanda 12

## 12.1 Domanda

Abibbo, Babibbo e Cabibbo hanno giocato una CTF da 36 challenge. Abibbo ha impiegato 6 ore per finirla, Babibbo ne ha impiegate 9 e Cabibbo 18. In quanto tempo l'avrebbero finita se avessero giocato tutti e 3 assieme?

*Nota: si suppone che la collaborazione tra i giocatori sia perfetta. Ad esempio se sia Abibbo che Babibbo avessero impiegato 10 ore, allora loro due assieme ne avrebbero impiegate solo 5.*

## 12.2 Risposte

(A) `2 ore`

(B) `3 ore`

(C) `4 ore`

(D) `5 ore`

## 12.3 Soluzione proposta

La risposta corretta è `(B) 3 ore`.

Calcoliamo, per ogni giocatore, quante challenge risolve in media all'ora:

$$\begin{aligned} \texttt{Abibbo:} &\quad 36\ \texttt{challenge}/6\ \texttt{ore} = 6\ \texttt{challenge/ora} \\ \texttt{Babibbo:} &\quad 36\ \texttt{challenge}/9\ \texttt{ore} = 4\ \texttt{challenge/ora} \\ \texttt{Cabibbo:} &\quad 36\ \texttt{challenge}/18\ \texttt{ore} = 2\ \texttt{challenge/ora} \end{aligned}$$

Quindi, se lavorassero assieme, risolverebbero $6 + 4 + 2 = 12$ challenge all'ora, finendo la CTF in solamente $36/12 = 3$ ore.