

Esercizio di analisi del codice

Leggi il codice e rispondi per iscritto alle domande sottostanti:

```
1. #!/bin/bash
2. x=0
3. y=0
4.
5. for z in "$@"; do
6.   if [[ $z -gt 0 ]]; then
7.     x=$((x + z))
8.     y=$((y + 1))
9.   fi
10. done
11.
12. if [[ $y -gt 0 ]]; then
13.   a=$((x / y))
14.   b=$((x % y))
15.   echo $a
16.
17. if [[ $b -ne 0 ]]; then
18.   echo $((($b / $y))
19. fi
20. fi
```

1. Dove sono dichiarate le variabili x e y? *all'inizio del codice*
2. Qual è il ruolo della variabile x nel codice? *è la somma di tutti gli arg. positivi*
3. Cosa rappresenta la variabile y e come viene modificata durante l'esecuzione? *è il contatore degli arg. positivi*
4. Cosa indica la variabile \$@? Qual è il suo 'tipo'? *è l'array di tutti gli arg.*
5. Quante volte viene eseguito il ciclo for? Quando si interrompe? *tante volte quanti sono gli arg.*
6. Che output si ottiene con input "2 6 3 -5 9 1"? *4 e 0*
7. Ci sono input per cui alcune righe di codice non vengono mai eseguite? Spiega e nel caso esemplifica. *Se tutti gli arg. sono negativi i corpi degli if non vengono eseguiti*
8. Cosa fa il blocco if dalla riga 17 alla riga 19? *Stampa il rapporto tra b e y se b è diverso da 0*
9. Quale valore viene memorizzato nella variabile z? Che nome potresti darle per chiarire il suo scopo? *è la variabile d'iterazione che salva l'elemento corrente dell'array*
10. Cosa fa il programma in generale e quale risultato produce in base agli input?
Il programma somma tutti i numeri positivi e ne stampa la media