

Sulla macchina virtuale fornita è già installato un “checker” che utilizzerete durante l’esame per verificare la correttezza degli esercizi di programmazione che saranno proposti.

Chi **NON** usa la macchina virtuale può installare il checker aprendo un terminale ed eseguendo il comando:

```
pip3 install dumbo-ae
```

Il checker si utilizza da terminale, come segue:

```
python3 -m dumbo_ae checker --of codice_esercizio soluzione.s
```

dove:

- **codice_esercizio** è il codice identificativo dell’esercizio di cui si vuole verificare la correttezza,
- **soluzione.s** è il file assembly contenente la soluzione dell’esercizio.

Ogni esercizio avrà quindi un codice diverso, che vi sarà indicato in sede d’esame.

Esercizio 1

Scaricare dal sito del corso “Lab10Esercizi.zip”. Al suo interno sono presenti due soluzioni dell’esercizio che chiedeva di calcolare la lunghezza della sequenza di Collatz di un numero letto da input (esercizio 5 di Lab 08):

- `collatz_ok.s` è una soluzione in cui l’output come il checker si aspetta,
- `collatz_no.s` è una soluzione in cui l’output non è come il checker si aspetta.

Eseguire quindi il checker con entrambe le soluzioni, spostandosi nella cartella contenente le due soluzioni e utilizzando il checker come segue:

```
python3 -m dumbo_ae checker --of cd80aa1e-f41e-4b0e-b4ac-b60b767e8aa9 collatz_ok.s
```

```
python3 -m dumbo_ae checker --of cd80aa1e-f41e-4b0e-b4ac-b60b767e8aa9 collatz_no.s
```

Il checker si aspetta in output soltanto un numero che rappresenta la lunghezza della sequenza di Collatz del numero letto da input. Si noti che quando la soluzione non produce l’output atteso, il checker riporta la percentuale di test con esito positivo e per ciascun input, riporta l’output effettivamente prodotto dalla soluzione testata (Actual output) e l’output atteso (expected output).

```
TEST OK: 0%
-----
      Input: 5
Actual output: Inserisci un numero positivo: Lunghezza 6
Expected output: 6
-----
      Input: 6
Actual output: Inserisci un numero positivo: Lunghezza 9
Expected output: 9
-----
```

Si noti inoltre che dal punto di vista del checker è indifferente che la stampa vada a capo o meno. Infatti con `collazt_ok.s` si ottiene che il 100% dei test è superato anche se la format string di stampa fosse:

```
fmt: .asciz "%d"
```

anziché:

```
fmt: .asciz "%d\n"
```

Esercizio 2

Realizzare un programma assembly che legga da input una sequenza di interi positivi terminata da un numero negativo e stampi soltanto la media intera dei numeri positivi letti. Se la sequenza fosse vuota, il programma non dovrebbe stampare nulla. Per svolgere l'esercizio si può partire dall'esercizio 1a di Lab 08, modificandolo opportunamente.

Dopo aver risolto l'esercizio, verificarne la correttezza con il checker utilizzando come codice:

```
10e31cc4-c9c0-4263-b4f3-fda8c2b95b5b
```

Se la sequenza fosse:

```
5 5 5 2 -6
```

Il programma dovrebbe stampare:

```
4
```

Se la sequenza fosse:

```
3 6 8 -1
```

Il programma dovrebbe stampare:

```
5
```

In Python il programma si può realizzare come segue:

```
somma = 0
conteggio = 0
numero = int(input())
while numero > 0:
    somma += numero
    conteggio += 1
    numero = int(input())
if conteggio > 0:
    media = somma // conteggio
    print(media)
```