

Esercizio 1: Assembly

Si consideri la seguente funzione iterativa, per $n \in \mathbb{N}$:

$$n_{i+1} = \begin{cases} n_i/2 & \text{se } n_i \text{ è pari,} \\ 3 \cdot n_i + 1 & \text{se } n_i \text{ è dispari.} \end{cases}$$

Dato un qualunque naturale $n_0 \geq 1$, si può trovare un numero finito k di iterazioni tali per cui $n_k = 1$.¹ Scrivere un programma che si comporta come segue:

1. Legge da tastiera un numero decimale su 8 bit. Sia tale valore n_0 .
2. Stampa, su righe distinte, tutti gli n_i con i da 1 a k .
3. Stampa il numero k di iterazioni corrispondenti al valore di n_0 immesso.
4. Termina.

Note:

- Non è necessario validare l'input oltre quanto già fatto dai sottoprogrammi `indecimal`
- Il valore k da trovare è il *primo* per cui $n_k = 1$.
- Il valore massimo di k così definito, per i possibili valori di n_0 considerati in questo esercizio, è < 255 .
- Si consiglia di utilizzare questo limite come controllo di sicurezza per evitare loop infiniti.
- Il valore massimo n_i raggiungibile, per tutti i possibili valori di n_0 considerati in questo esercizio, è $13120 < n_i < 40000$.

Le stampe vanno formattate come da gli esempi di output in .txt allegati al link:

<https://tinyurl.com/jnxrhhjc>

Si ponga attenzione alla formattazione di questi file, che fa parte delle specifiche.

La documentazione Assembler in formato PDF è scaricabile al link: <https://tinyurl.com/ys9euy9>

¹Questa è nota come *congettura* di Collatz. È certamente vera per tutti i valori di n_0 considerati in questo esercizio.