

Esercizio 1: Assembly

Scrivere un programma che dopo aver letto un numero naturale su 32 bit, ne stampa i byte in notazione esadecimale, prima in ordine *big-endian* e poi in ordine *little-endian*.

Più precisamente, il programma si comporta come segue:

1. Legge un numero naturale n su 32 bit.
2. Stampa, in notazione esadecimale, la rappresentazione di n come serie *big-endian* di byte, in notazione esadecimale e separati da spazi.
3. Stampa, in notazione esadecimale, la rappresentazione di n come serie *little-endian* di byte, in notazione esadecimale e separati da spazi.
4. Termina.

Note:

- Per la lettura di numeri, è sufficiente la validazione fatta dai sottoprogrammi `indecimal`.
- Per la stampa di rappresentazioni esadecimali, si utilizzino i sottoprogrammi `outbyte` / `outword` / `outlong`.
- È particolarmente apprezzato, ma non necessario, la definizione e l'utilizzo di appropriati sottoprogrammi.

Le stampe vanno formattate come da gli esempi allegati al link: <https://tinyurl.com/yc7h3pfj>

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/mryny9jt>