

## Esercizio 1: Assembly

Scrivere un programma che si comporta come segue:

1. legge con eco da tastiera una stringa di al più 30 caratteri ASCII. Se la stringa inserita è di 0 caratteri, termina.
2. legge con eco da tastiera due numeri decimali  $A$  e  $B$
3. interpreta  $A$  e  $B$  come indici a base 0 che delimitano una sottostringa della stringa inserita, dove  $A$  indica la posizione del primo carattere della sottostringa e  $B$  indica la posizione successiva a quella dell'ultimo carattere della sottostringa. Se tali indici sono validi e la sottostringa risultante è di lunghezza non nulla, prosegue al passo 4. Altrimenti termina
4. Data la sottostringa delimitata dagli indici  $A$  e  $B$ , stampa su righe distinte:
  - la sottostringa
  - l'inverso della sottostringa, ossia la stessa letta da destra verso sinistra
  - "PALINDROMA" se la sottostringa è *palindroma*, "NON PALINDROMA" altrimenti
5. lascia una riga bianca e ritorna al punto 2

Note:

- Non è necessario validare l'input oltre quanto già fatto dai sottoprogrammi inline e indecimal
- Nel valutare la lunghezza della stringa inserita, così come i caratteri appartenenti a una sottostringa, non vanno considerati eventuali caratteri di terminazione
- Una stringa si dice *palindroma* se è uguale sia letta da sinistra verso destra sia da destra verso sinistra. Non si faccia alcuna distinzione particolare per maiuscole/minuscole, spazi o altro

Un esempio di output è allegato in formato .txt .

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