

## CARTA ORO

1)

Scrivere un programma in Java che legge da tastiera una stringa che deve avere il seguente formato:

una prima sequenza di cifre che rappresenta un numero naturale

seguita da

carattere + oppure carattere –

seguito da

un'altra sequenza di cifre che rappresenta un altro numero naturale

...

...

fino ad un'ultima sequenza di cifre che rappresenta un ultimo naturale.

Se la stringa letta non rispetta il formato richiesto il programma termina con un messaggio di errore. Se invece lo rispetta, il programma deve calcolare, stampandolo a video, il valore dell'espressione:

numero rappresentato dalla I sequenza di cifre

+ (oppure -, a seconda del carattere che appare nella stringa)

numero rappresentato dalla II sequenza di cifre

+ (oppure -, a seconda del carattere che appare nella stringa)

...

numero rappresentato dall'ultima sequenza di cifre

Esempio. Se la stringa letta è

"101+14-31+47-59-74"

il programma deve calcolare il valore

-2

che corrisponde al valore dell'espressione aritmetica

101+14-31+47-59-74

2)

Scrivere un programma in Java che legge da tastiera due stringhe (non necessariamente lunghe uguali) la prima contenente solo lettere dell'alfabeto minuscole, la seconda solo lettere dell'alfabeto maiuscole. Se le stringhe lette non rispettano quanto richiesto il programma termina con un messaggio di errore. Altrimenti il programma stampa a video tutte le possibili coppie che si ottengono abbinando un carattere della prima stringa ad un carattere della seconda stringa.

3)

Scrivere un programma in Java che legge da tastiera una stringa composta da lettere minuscole dell'alfabeto e due numeri naturali  $m$  e  $n$  compresi tra 0 e  $k-1$ , dove  $k$  è la lunghezza della stringa. Qualora l'utente, una volta inserita una stringa, inserisse un valore maggiore della lunghezza della stringa, oppure la stringa inserita non fosse composta solamente da lettere dell'alfabeto minuscole, il programma termina stampando messaggio d'errore. Il programma deve costruire una nuova stringa che si ottiene da quella inserita convertendo in maiuscolo le lettere che si trovano in posizione

$m$

$(m+n) \bmod k$

$(m+2n) \bmod k$

$(m+3n) \bmod k$

...

[cercare di capire quante volte al massimo procedere prima di reincontrare una posizione già visitata e quindi interrompere la sequenza]