

Laboratorio di Programmazione

(Corso di Laurea in Informatica)

APPELLO del 12 Febbraio 2018

Avvertenze

- I programmi realizzati DEVONO rispettare le specifiche fornite sull'input e sulla forma dell'output, senza aggiungere messaggistica ulteriore o utilizzare termini e/o spaziatura differenti. Parte della valutazione è infatti sulla comprensione e sul rispetto delle specifiche.
 - Nello svolgimento dell'elaborato è possibile usare qualunque classe delle librerie standard di Java.
 - Si raccomanda di salvare, compilare e fare upload delle soluzioni con una certa regolarità. In ogni caso NON sarà riaperta la sessione di upload per chi non ha consegnato in tempo.
 - Per la procedura di **consegna** si veda in fondo al documento.
-

1 Triangoli

1.1 Descrizione

Dati tre segmenti qualsiasi a , b , c , è possibile costruire un triangolo se e solo se la lunghezza di ciascuno è strettamente minore della somma degli altri due (un triangolo “piatto” non è più un triangolo).

Scrivete un programma (una classe pubblica “Triangoli” dotata del metodo `main`) che, dati 6 parametri interi sulla linea di comando, determina se sia i primi tre che i secondi tre corrispondono alle dimensioni di un triangolo. Se sì, il programma determina se i due triangoli sono uguali (hanno le stesse dimensioni).

Se i parametri sono negativi, non sono numerici o non sono sei, il programma stampa il messaggio “Dati non validi” e termina.

La classe deve essere dotata dei seguenti metodi:

- `boolean isTriangolo(int a, int b, int c)`, che restituisce `true` se i parametri corrispondono a lati che possono formare un triangolo, `false` altrimenti.
- `boolean areEquals (int[] triangolo1, int[] triangolo2)`, che restituisce `true` se i parametri corrispondono a triangoli con le stesse dimensioni, `false` altrimenti.

1.2 Esempi

```
$ java Triangoli 3 4 5 4 5 3
[3, 4, 5] è un triangolo
[4, 5, 3] è un triangolo
Sono uguali
```

```
$ java Triangoli 3 4 5 4 5 6  
[3, 4, 5] è un triangolo  
[4, 5, 6] è un triangolo  
Non sono uguali
```

```
$ java Triangoli 3 4 5 4 7 15  
[3, 4, 5] è un triangolo  
[4, 7, 15] non è un triangolo
```

```
$ java Triangoli a b c  
Dati non validi
```

2 Vigenere

2.1 Descrizione

Alcuni metodi di cifratura di parole (ad esempio quello di Cesare) sono basati sull'operazione di *rotazione* di una lettera. Ruotare una lettera c maiuscola di un valore rot significa traslare c di rot posti, ripartendo da 'A' quando si arriva alla 'Z'.

Indicando con

$$c + rot = cRot$$

il fatto che ruotando la lettera c di rot posti si ottiene la lettera $cRot$, si ha:

$$A + 1 = B$$

$$A + 2 = C$$

$$D + 2 = F$$

$$Y + 1 = Z \quad // \text{ Y è la penultima lettera dell'alfabeto}$$

$$Y + 2 = A$$

Nel cifrario di Cesare la chiave è una lettera e la lettera cifrata viene ottenuta ruotando la lettera in chiaro di un numero di posti corrispondenti alla posizione della lettera chiave nell'alfabeto (0 per A, 1 per B, ...).

Il cifrario di Vigenère è un metodo di cifratura che consiste in una generalizzazione del cifrario di Cesare: invece di ruotare sempre dello stesso numero di posti la lettera da cifrare, questa viene ruotata di un numero di posti variabile, determinato in base ad una parola chiave. Più precisamente, data una parola chiave (detta anche *verme*) di lunghezza n , la prima lettera del testo da cifrare viene cifrata con la prima lettera del verme, la seconda del testo con la seconda del verme e così via fino alla n -esima; con la lettera $n + 1$ -esima del testo si riprende a utilizzare il verme dalla prima lettera e così via fino alla fine del testo.

Ad esempio:

Testo in chiaro - RAPPORTOIMMEDIATO

Verme - VERME

Testo cifrato - MEGBSMXFUQHIUUEOS

Come potete notare, la R all'inizio del testo e la R in posizione $n + 1$ (qui il verme è di 5 lettere, quindi in sesta posizione) vengono cifrate nello stesso modo, ma le due P no.

Scrivete un programma (una classe pubblica "Vigenere" dotata del metodo `main`) che, data una stringa s come parametro sulla linea di comando, legge un testo da standard input e, dopo aver trasformato in MAIUSCOLO sia la chiave che il testo da cifrare, emette su standard output il testo cifrato utilizzando la stringa s come chiave (verme) e cifrando solo le lettere, non le cifre e non tutti gli altri caratteri (punteggiatura, ecc.), che verranno trascritti come sono.

Se manca il parametro su linea di comando, il programma stampa il messaggio "Chiave non fornita" e termina.

2.2 Esempio 1

Il testo:

AA, A. AAAA A 12 AAA

aaaa: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

aa

Con “verme”:

VERMISSIMO

Diventa:

VE, R. MISS I 12 MOV

ERMI: SSIMOVERMISSIMOVERMISSIMOVERMI
SS

2.3 Esempio 2

Il testo:

Lorem ipsum dolor sit amet, consectetur

adipiscing elit, sed do eiusmod tempor incididunt

ut labore et dolore magna aliqua.

Ut enim ad minim veniam, quis nostrud exercitation ullamco

laboris nisi ut aliquip ex ea commodo consequat.

Duis aute irure dolor in reprehenderit in

voluptate velit esse cillum dolore eu fugiat nulla pariatur.

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui
officia deserunt mollit anim id est laborum.

Con “verme”:

vermenormaleminuscolo

Diventa:

GSIIQQ VDJGM OSXWE MAV OXSO, GFZWRQKQTFV

MLVJAUQTBB ICUX, FSU PO PMGAZIV VSXDJV ZZGVRZPUYX

GB YUTQFP SO HFXSES DMGYE MTVKMC.

IE SIMD MH ZWEUM GIZQNG, IWWD BJWKDYQ SOQRNMFICGCP IWZVQTA

PNPFDID RUAV OL CZTEPMG QB RO TAMXSPW PIFUSBIVX.

UGMF OLFE TVGZR XGNCC WI VVBVRVVZDPVUB VH

NQZFDOEKQ ZRZZF EDWQ KVFDWA OCGSIQ IH TLSILX ZCYFS ROCWVXLD.

IKQVBTPYD AVHL QQNOZGRF GHDZPAEEF VBH HTCTRZRK, EYAH ZZ CFPBI DOA
QTQWXR PIFSIGNE QATYCL CBTA DH VEX YOSARFQ.

3 Andamento di una funzione

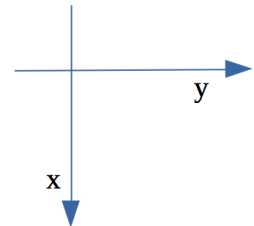
3.1 Descrizione

Scrivete un programma (una classe pubblica “FunPlot” dotata del metodo `main`) che calcoli l’andamento di una funzione data in un certo intervallo. In particolare il programma, dati i seguenti parametri su linea di comando:

- i due estremi dell’intervallo delle ascisse da considerare (x_{min} e x_{max})
- la distanza sulle ascisse tra due punti consecutivi (il *passo*)
- una stringa tra queste: `square`, `3x2`, `resto10`, `pow4` e `tan`, che rappresentano il nome di una funzione,

per ogni coppia di valori corrispondenti x_i , y_i deve visualizzare su standard output, utilizzando il carattere “*”, il grafico della funzione per punti.

Si disegni il grafico uno punto per riga, con l’asse delle ordinate orientato orizzontalmente da sinistra a destra e quello delle ascisse orientato verticalmente dall’alto verso il basso, cioè ruotato di 90° rispetto alla convenzione usuale, per adattarsi alla naturale direzione di stampa su standard output (vedi figura).



Per ogni punto, il formato dell’output è: il valore x_i , seguito da “: ” (due punti e spazio) e da un asterisco posizionato in modo da rappresentare il valore della funzione in quel punto, seguito dal valore y_i (vedi esempio più avanti).

Verranno rappresentati con un asterisco solo i valori y_i nell’intervallo $(-40, 40)$. Se il valore y_i non appartiene all’intervallo $(-40, 40)$, non verrà visualizzato l’asterisco, ma solo il valore y_i subito dopo i “: ”.

Nota: la posizione di “*” approssima il valore della funzione all’intero più vicino in questo modo:

- se ho una funzione che vale -39.00 per $x = 1$, il programma visualizzerà:

```
1.00:  * -39.00
```

cioè l’asterisco a destra di uno spazio rispetto a “: ”;

- se la funzione vale -35.2 per $x = 1$, il programma visualizzerà:

```
1.00:      * -35.20
```

cioè l’asterisco a destra di cinque spazi rispetto a “: ”, e così via.

Le funzioni da considerare sono:

- la funzione $y = x^2$, rappresentata dalla stringa `square`.
- la funzione $y = 3x + 2$, rappresentata dalla stringa `3x2`.
- la funzione resto intero (o modulo) della divisione per 10, rappresentata dalla stringa `resto10`.
- la funzione $y = x^4$, rappresentata dalla stringa `pow4` (si veda la funzione `pow` della classe `Math`).
- la funzione tangente, rappresentata dalla stringa `tan` e disponibile nella classe `Math` con quello stesso nome.

Se i parametri su linea di comando non sono compatibili con le specifiche, il programma stampa il messaggio “Dati non validi” e termina.

Per stampare un numero x con due cifre decimali potete usare `System.out.printf("%.2f", x)`.

3.2 Esempio

```
$ java FunPlot 0.0 5.0 0.2 tan
0.00:          * 0.00
0.20:          * 0.20
0.40:          * 0.42
0.60:          * 0.68
0.80:          * 1.03
1.00:          * 1.56
1.20:          * 2.57
1.40:          * 5.80
1.60:    * -34.23
1.80:          * -4.29
2.00:          * -2.19
2.20:          * -1.37
2.40:          * -0.92
2.60:          * -0.60
2.80:          * -0.36
3.00:          * -0.14
3.20:          * 0.06
3.40:          * 0.26
3.60:          * 0.49
3.80:          * 0.77
4.00:          * 1.16
4.20:          * 1.78
4.40:          * 3.10
4.60:          * 8.86
4.80:          * -11.38
```

Consegna

Si ricorda che le classi devono essere tutte *public* e che vanno consegnati tutti (e soli) i file *.java* prodotti. NON vanno consegnati i *.class*. Per la consegna, eseguite l'upload dei SINGOLI file sorgente (NON un file archivio!) dalla pagina web: <http://upload.di.unimi.it> nella sessione che vi è stata indicata.

*** ATTENZIONE!!! ***

NON VERRANNO VALUTATI GLI ESERCIZI CON ERRORI DI COMPILAZIONE O LE CONSEGNE CHE NON RISPETTANO LE SPECIFICHE (ad esempio consegnare un archivio zippato è sbagliato)

Per ritirarsi fare l'upload di un file vuoto di nome `ritirato.txt`.
