Informazione

Autore: **Daniil Radchanka**

Email: **daniil.radchanka@unifi.it**

Matricola: **7079901**

Data di consegna: **05.01.2023**

Descrizione

Il programma inizia il suo funzionamento dalla funzione *main*, dove scannerizzando caratteri di *mycypher* applichiamo corrispondente algoritmo di cifratura. Dopo di aver applicato ogni algoritmo di cifratura si procede in ordine inverso applicando algoritmi di decifratura.

La variabile *encryptedtext* e molto importante per il corretto funzionamento del codice perché deve essere dichiarata ad ultimo in *.data*. Questa variabile cambierà la sua lunghezza dopo l'applicazione di cifratura e decifratura per occorrenze e se non sarà dichiarata per ultimo tal funzioni di cifratura scriveranno i suoi risultati nei prossimi variabili dopo di e *encryptedtext.*

In generale si può suddividere tutti le funzioni in due categorie:

1. Funzioni principali (main, ogni funzione di cifratura e decifratura)
2. Funzioni utili (trasformazione di un numero in stringa, controllo del tipo di carattere e etc.)

Quasi tutti le funzioni principali sono fatte in maniera descritta nel file del progetto per AE, tramite una in quale voglio fare la attenzione dell'implementazione mia. La funzione di cifratura e decifratura per occorrenze aveva la difficolta perché era l'unica che non "spostava" il carattere per una formula, ma aveva una nuova interpretazione per ogni carattere incontrato mentre si applicava tal algoritmo.

La mia idea di cifratura per occorrenze era seguente:

- Allochiamo la stringa risultante subito dopo la stringa originale (encryptedtext).
- Per ogni carattere troviamo suoi prossimi occorrenze e marchiamo tal occorrenze in qualche modo per sapere che questi caratteri erano già scelti per la cifratura.
- Per ogni occorrenza di un carattere salviamo la sua posizione in riga risultante con il simbolo "-" precedente.
- Quanto tutti occorrenze erano salvati proseguiamo con il carattere scelto e mettiamo lo spazio in stringa risultante.
- Quando i caratteri della parte originale sono finiti copiamo la stringa risultane in posizione nella stringa originale.

La mia idea di decifratura per occorrenze era seguente:

- Allochiamo la riga risultante subito dopo la riga originale (encryptedtext).

- Ricordiamo il carattere in stringa originale per poter metterla nella stringa risultante.
- Scannerizziamo la stringa finché non si trova lo spazio e ogni volta trovato "-" prendiamo il numero seguente dopo "-", convertiamo tal numero dalla stringa a un numero e scriviamo il carattere salvato in posizione di tal numero.
- Dopo di aver preso tutti i numeri del carattere si prosegue con il carattere che viene subito dopo lo spazio.
- Alla fine copiamo la stringa risultante in posizione della stringa originale.

Dal punto di vista della memoria e l'uso dei registri tramite l'uso classico per invocazione di un metodo si deve osservare solo l'importanza della posizione di encryptedtext, che deve essere obbligatoriamente dichiarato per ultimo.

Test di corretto funzionamento

Test 1

Data

```
9
0 mycypher: .string "ABCDE"
1 myplaintext: .string "Tests"
2 sostK: .word -108
3 blockKey: .string "aA-Bb/"
4
```

Result

```
Tests
Paopo
qb\r1
q-1 b-2 \-3 r-4 1-5
J-8 Y-7 \-6 I-5 8-4
4-8 5-I 6-\ 7-Y 8-J
J-8 Y-7 \-6 I-5 8-4
q-1 b-2 \-3 r-4 1-5
qb\r1
Paopo
Tests
```

Test 2

Data

```
mycypher: .string "AABBCCDDE"
myplaintext: .string "te/sT"
sostK: .word -108
blockKey: .string "aB-"
```

Result

```
te/sT
pa/oP
lw/kL
-y|,N
N{iMP
N-1 {-2 i-3 M-4 P-5
N-1 --2-6-10-14-18 1-3  -4-8-12-16 {-5 2-7 i-9 3-11 M-13 4-15 P-17 5-19
m-8 --7-3-89-85-81 8-6  -5-1-87-83 {-4 7-2 R-0 6-88 n-86 5-84 k-82 4-80
N-1 --2-6-10-14-18 1-3  -4-8-12-16 {-5 2-7 i-9 3-11 M-13 4-15 P-17 5-19
91-5 71-P 51-4 31-M 11-3 9-i 7-2 5-{ 61-21-8-4-  3-1 81-41-01-6-2-- 1-N
N-1 --2-6-10-14-18 1-3  -4-8-12-16 {-5 2-7 i-9 3-11 M-13 4-15 P-17 5-19
m-8 --7-3-89-85-81 8-6  -5-1-87-83 {-4 7-2 R-0 6-88 n-86 5-84 k-82 4-80
N-1 --2-6-10-14-18 1-3  -4-8-12-16 {-5 2-7 i-9 3-11 M-13 4-15 P-17 5-19
N-1 {-2 i-3 M-4 P-5
N{iMP
-y|,N
lw/kL
pa/oP
te/sT
```

Test 3

Data

```
10 mycypher: .string "EDCBA"
11 myplaintext: .string "simple test -1"
12 sostK: .word 5
13 blockKey: .string "104A-"
```

Result

```
simple test -1
1- tset elpmis
8- GHVG VOKNRH
8-1 --2  -3-8 G-4-7 H-5-14 V-6-9 O-10 K-11 N-12 R-13
)}%!z~"t!z$},!4~$!8m9}).~%pJ.#~)tPz" tLz"!tOz""tSz"#
)}%!e~"y!e$},!4~$!8r9}).~%uO.#~)yUe" yQe"!yTe""yXe"#
)}%!z~"t!z$},!4~$!8m9}).~%pJ.#~)tPz" tLz"!tOz""tSz"#
8-1 --2  -3-8 G-4-7 H-5-14 V-6-9 O-10 K-11 N-12 R-13
8- GHVG VOKNRH
1- tset elpmis
simple test -1
```