

# 14BHD INFORMATICA, A.A. 2020/2021

## Esercitazione di Laboratorio 10

---

### Obiettivi dell'esercitazione

- Lettura e scrittura di file
- Acquisizione ed elaborazione di dati reali
- Gestire errori di input/output mediante eccezioni

### Contenuti tecnici

- Leggere e scrivere file di testo
  - Elaborare raccolte dati
- 

### Da risolvere in laboratorio

Esercizio 1. Scrivete un programma che legga il file di testo *input.txt*. Ogni riga letta va scritta nel file *output.txt* preceduta dal numero di riga, inserita come commento tra caratteri `/*` e `*/`. Se, ad esempio, il file *input.txt* fosse il seguente:

```
Enola Gay
è il bombardiere che il 6 agosto 1945,
sganciò su Hiroshima la prima bomba atomica
soprannominata Little Boy.
```

Il file *output.txt* generato sarebbe:

```
/*1*/Enola Gay
/*2*/è il bombardiere che il 6 agosto 1945,
/*3*/sganciò su Hiroshima la prima bomba atomica
/*4*/soprannominata Little Boy.
```

[P7.2]

Esercizio 2. Scrivete un programma che legga tutte le righe di un file di testo (*input.txt*), ne inverta l'ordine e le scriva in un altro file (*output.txt*). Ad esempio, se il file *input.txt* contiene queste righe:

```
Mary had a little lamb
Its fleece was white as snow
And everywhere that Mary went
The lamb was sure to go.
```

allora il file *output.txt* conterrà:

```
The lamb was sure to go.  
And everywhere that Mary went  
Its fleece was white as snow  
Mary had a little lamb
```

[P7.9]

- Esercizio 3. Scrivere un programma che cerchi una data parola nel contenuto di un gruppo di file. Il programma innanzitutto chiede l'elenco dei file (da inserire su una sola riga, separati da virgole e la parola da cercare. I nomi dei file saranno memorizzati in una lista (*files*), la parola da cercare viene memorizzata in una variabile. Occorre visualizzare tutte le righe che contengono la parola, indipendentemente da maiuscole o minuscole, ciascuna riga preceduta dal nome del file in cui si trova. Ad esempio, se la parola fosse "ring", e lista contenesse:

```
book.txt, address.txt, homework.py
```

allora il programma potrebbe visualizzare quanto segue:

```
book.txt: There is only one Lord of the Ring, only one who can bend  
it to his will  
book.txt: The ring has awoken; it's heard its masters call.  
address.txt: Kris Kringle, North Pole  
address.txt: Homer Simpson, Springfield  
homework.py: string = "text"
```

La parola da cercare viene inserita dall'utente e sono valide anche le parole che contengono quella da cercare (es. ring e ringhiare). [P7.6]

- Esercizio 4. Il responsabile amministrativo di un albergo registra le vendite in un file di testo. Ogni riga contiene le seguenti 4 informazioni, separate da caratteri "punto e virgola": il nome del cliente, il servizio venduto (ad esempio, cena, conferenza, alloggio, e così via), l'importo pagato e la data dell'evento. Scrivete un programma che legga un tale file di testo e visualizzi l'importo totale relativo a ciascun tipo di servizio, segnalando un errore se il file non esiste oppure se il suo formato non è corretto (verificando cioè che ci siano 4 campi per riga e che il prezzo sia float). [P7.29]
-

Da risolvere a casa

Esercizio 5. *Cifratura monoalfabetica casuale.* Il cifrario di Cesare, che trasla ogni lettera di una quantità fissa, è troppo facile da violare. Ecco un'idea migliore: come chiave, invece di usare un numero, usiamo una parola, che immaginiamo essere FEATHER. Per prima cosa eliminiamo le lettere duplicate dalla parola chiave, ottenendo FEATHR, poi aggiungiamo in fondo ad essa le altre lettere dell'alfabeto, in ordine inverso:

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| F | E | A | T | H | R | Z | Y | X | W | V | U | S | Q | P | O | N | M | L | K | J | I | G | D | C | B |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

Ora cifriamo le lettere, seguendo questo schema:

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ |
| F | E | A | T | H | R | Z | Y | X | W | V | U | S | Q | P | O | N | M | L | K | J | I | G | D | C | B |

Tutti gli altri caratteri (spazi, cifre, punteggiatura, ...) devono rimanere invariati. Scrivete un programma che cifri o decifri un file di testo (il cui nome è inserito dall'utente) usando una parola chiave, inserita dall'utente e memorizzata in una variabile, e scrivendo le informazioni decifrate in un file di testo scelto dall'utente.

[P7.20]

Esercizio 6. Scrivere un programma che visualizzi l'elenco degli esami superati da uno studente, con i relativi voti. È disponibile un file, *classes.txt*, che contiene i nomi di tutti gli insegnamenti erogati nell'istituto scolastico (un college statunitense), il cui contenuto sarà simile a questo:

CSC1  
CSC2  
CSC46  
CSC151  
MTH121  
...

Poi, per ogni insegnamento, è disponibile un file (il cui nome è pari al codice dell'insegnamento seguito da *.txt*) che elenca gli studenti che hanno superato il relativo esame e contiene i numeri identificativi degli studenti (ID) e i voti, come questo, che potrebbe essere il file *CSC2.txt*:

11234 A-

12547 B  
16753 B+  
21886 C  
...

Scrivere un programma che chieda all'utente l'identificativo (ID) di uno studente e visualizzi l'elenco degli esami che tale studente ha superato, con i relativi voti ottenuti, come in questo esempio:

Student ID 16753  
CSC2 B+  
MTH121 C+  
CHN1 A  
PHY50 A-

[P7.28]

Esercizio 7. *Cifrario di Playfair*. Un diverso modo per impedire la decifrazione di un testo mediante la semplice analisi delle frequenze delle singole lettere consiste nel cifrare insieme coppie di lettere e un semplice schema per farlo è il cifrario di Playfair. Si sceglie una parola chiave e vi si eliminano le lettere duplicate, poi la si inserisce, seguita ordinatamente dalle altre lettere dell'alfabeto inglese (non presenti dunque nella parola chiave), in una scacchiera  $5 \times 5$  (dato che ci sono soltanto 25 caselle e l'alfabeto inglese ha 26 lettere, la I e la J sono considerate indistinguibili. Verificare che nel testo sorgente le lettere siano pari, sennò aggiungere una "Z"). Ecco lo schema che si ottiene usando PLAYFAIR come parola chiave:

|   |   |   |   |   |
|---|---|---|---|---|
| P | L | A | Y | F |
| I | R | B | C | D |
| E | G | H | K | M |
| N | O | Q | S | T |
| U | V | W | X | Z |

Per cifrare una coppia di lettere, ad esempio AM, si cerca il rettangolo che ha A e M negli angoli.

|   |   |   |   |   |
|---|---|---|---|---|
| P | L | A | Y | F |
| I | R | B | C | D |
| E | G | H | K | M |
| N | O | Q | S | T |
| U | V | W | X | Z |

La codifica di questa coppia si ottiene prendendo gli altri due angoli del rettangolo, in questo caso FH. Se le due lettere della coppia si trovano sulla stessa riga o sulla stessa colonna, come GO, vengono semplicemente scambiate tra loro. La decifrazione avviene nello stesso modo. Scrivete un programma che cifri o decifri un file di testo secondo lo schema qui presentato. [P7.23]