

Esame 20220901

Esercizio 1

(1) Esercizio 1 v1

ESSAY marked out of 10 penalty 0 File picker

Scrivere nel file `eserciziol.cc` un programma che, presi come argomento del main:

- Il nome di un file di testo in input, contenente una o più righe.
- Il nome di un file di testo da usare per l'output.

legga il contenuto del file di input e scriva nel file di output la lista delle dimensioni di ciascuna riga del file in input. Viene inoltre richiesto di stampare alla fine del file di output la riga più **lunga** e la sua **dimensione**.

Supponiamo che il primo file `input.txt` contenga

```
C++ feels like a new language.  
That is, I can express my ideas more clearly,  
more simply, and more directly in C++11  
than I could in C++98. Furthermore, the resulting  
programs are better checked by the compiler  
and run faster.  
Bjarne Stroustrup, The C++ Programming Language (4th Edition)
```

allora il comando

```
./eserciziol.out input.txt output.txt
```

e dovrà produrre un file chiamato `output.txt` che conterrà i seguenti valori:

```
30  
45  
39  
49  
43  
15  
61  
Bjarne Stroustrup, The C++ Programming Language (4th Edition) 61
```

Note:

- Per dimensione di una riga si intende il numero dei caratteri ivi contenuti, escluso il carattere "a capo". Spazi e segni di punteggiatura vanno conteggiati nel computo di ogni riga.
- Si assuma che ogni riga contenga al massimo 255 caratteri.
- Non è concesso fare assunzioni sul numero **massimo** di righe presenti nel file pena l'annullamento dell'esercizio.
- Non è ammesso l'uso di alcuna routine di gestione delle stringhe (ad esempio `strlen` o `strcpy`).

- È consentito l'utilizzo della funzione `getline (char* s, streamsize n)`. Ad esempio, `input.getline(buffer, 10)`.
- Non è consentito l'utilizzo di altre funzioni di libreria "particolari" diverse da quelle specificate sopra o da quelle standard necessarie a risolvere l'esercizio.
- Le uniche assunzioni che si possono fare sull'input e su dimensioni di eventuali strutture/array utilizzate nel file di partenza fornito sono **solo quelle espressamente specificate in questo testo** (e NON quelle riportate nel file fornito, che sono SOLO indicative per consentire di svolgere l'esame).
- All'interno di questo programma **non è ammesso** l'utilizzo di variabili globali o di tipo `static`.
- Si ricorda che, l'esempio di esecuzione è puramente indicativo, e la soluzione proposta NON deve funzionare solo per l'input fornito, ma deve essere robusta a variazioni compatibili con la specifica riportata in questo testo.

Information for graders:

(2) Esercizio 1 v2

ESSAY

marked out of 10

penalty 0

File picker

Scrivere nel file `eserciziol.cc` un programma che, presi come argomento del main:

- Il nome di un file di testo in input, contenente una o più righe.
- Il nome di un file di testo da usare per l'output.

legga il contenuto del file di input e scriva nel file di output la lista delle dimensioni di ciascuna riga del file in input. Viene inoltre richiesto di stampare alla fine del file di output la riga più **corta** e la sua **dimensione**.

Supponiamo che il primo file `input.txt` contenga

```
C++ feels like a new language.  
That is, I can express my ideas more clearly,  
more simply, and more directly in C++11  
than I could in C++98. Furthermore, the resulting  
programs are better checked by the compiler  
and run faster.  
Bjarne Stroustrup, The C++ Programming Language (4th Edition)
```

allora il comando

```
./eserciziol.out input.txt output.txt
```

e dovrà produrre un file chiamato `output.txt` che conterrà i seguenti valori:

```
30  
45  
39  
49  
43  
15  
61  
and run faster. 15
```

Note:

- Per dimensione di una riga si intende il numero dei caratteri ivi contenuti, escluso il carattere "a capo". Spazi e segni di punteggiatura vanno conteggiati nel computo di ogni riga.
- Si assuma che ogni riga contenga al massimo 255 caratteri.
- Non è concesso fare assunzioni sul numero **massimo** di righe presenti nel file pena l'annullamento dell'esercizio.
- Non è ammesso l'uso di alcuna routine di gestione delle stringhe (ad esempio `strlen` o `strcpy`).
- È consentito l'utilizzo della funzione `getline (char* s, streamsize n)`. Ad esempio, `input.getline(buffer, 10)`.
- Non è consentito l'utilizzo di altre funzioni di libreria "particolari" diverse da quelle specificate sopra o da quelle standard necessarie a risolvere l'esercizio.

- Le uniche assunzioni che si possono fare sull'input e su dimensioni di eventuali strutture/array utilizzate nel file di partenza fornito sono **solo quelle espressamente specificate in questo testo** (e NON quelle riportate nel file fornito, che sono SOLO indicative per consentire di svolgere l'esame).
- All'interno di questo programma **non è ammesso** l'utilizzo di variabili globali o di tipo `static`.
- Si ricorda che, l'esempio di esecuzione è puramente indicativo, e la soluzione proposta NON deve funzionare solo per l'input fornito, ma deve essere robusta a variazioni compatibili con la specifica riportata in questo testo.

Information for graders:

(3) Esercizio 1 v3

ESSAY

marked out of 10

penalty 0

File picker

Scrivere nel file `eserciziol.cc` un programma che, presi come argomento del main:

- Il nome di un file di testo in input, contenente una o più righe.
- Il nome di un file di testo da usare per l'output.

legga il contenuto del file di input e scriva nel file di output la lista delle dimensioni di ciascuna riga del file in input. Viene inoltre richiesto di stampare alla fine del file di output la **prima riga** del file di input e la **dimensione di quella più corta**.

Supponiamo che il primo file `input.txt` contenga

```
C++ feels like a new language.  
That is, I can express my ideas more clearly,  
more simply, and more directly in C++11  
than I could in C++98. Furthermore, the resulting  
programs are better checked by the compiler  
and run faster.  
Bjarne Stroustrup, The C++ Programming Language (4th Edition)
```

allora il comando

```
./eserciziol.out input.txt output.txt
```

e dovrà produrre un file chiamato `output.txt` che conterrà i seguenti valori:

```
30  
45  
39  
49  
43  
15  
61  
C++ feels like a new language. 15
```

Note:

- Per dimensione di una riga si intende il numero dei caratteri ivi contenuti, escluso il carattere "a capo". Spazi e segni di punteggiatura vanno conteggiati nel computo di ogni riga.
- Si assuma che ogni riga contenga al massimo 255 caratteri.
- Non è concesso fare assunzioni sul numero **massimo** di righe presenti nel file pena l'annullamento dell'esercizio.
- Non è ammesso l'uso di alcuna routine di gestione delle stringhe (ad esempio `strlen` o `strcpy`).
- È consentito l'utilizzo della funzione `getline (char* s, streamsize n)`. Ad esempio, `input.getline(buffer, 10)`.
- Non è consentito l'utilizzo di altre funzioni di libreria "particolari" diverse da quelle specificate sopra o da quelle standard necessarie a risolvere l'esercizio.

- Le uniche assunzioni che si possono fare sull'input e su dimensioni di eventuali strutture/array utilizzate nel file di partenza fornito sono **solo quelle espressamente specificate in questo testo** (e NON quelle riportate nel file fornito, che sono SOLO indicative per consentire di svolgere l'esame).
- All'interno di questo programma **non è ammesso** l'utilizzo di variabili globali o di tipo `static`.
- Si ricorda che, l'esempio di esecuzione è puramente indicativo, e la soluzione proposta NON deve funzionare solo per l'input fornito, ma deve essere robusta a variazioni compatibili con la specifica riportata in questo testo.

Information for graders:

(4) Esercizio 1 v4

ESSAY marked out of 10 penalty 0 File picker

Scrivere nel file `eserciziol.cc` un programma che, presi come argomento del main:

- Il nome di un file di testo in input, contenente una o più righe.
- Il nome di un file di testo da usare per l'output.

legga il contenuto del file di input e scriva nel file di output la lista delle dimensioni di ciascuna riga del file in input. Viene inoltre richiesto di stampare alla fine del file di output la **prima riga** del file di input e la **dimensione di quella più lunga**.

Supponiamo che il primo file `input.txt` contenga

```
C++ feels like a new language.  
That is, I can express my ideas more clearly,  
more simply, and more directly in C++11  
than I could in C++98. Furthermore, the resulting  
programs are better checked by the compiler  
and run faster.  
Bjarne Stroustrup, The C++ Programming Language (4th Edition)
```

allora il comando

```
./eserciziol.out input.txt output.txt
```

e dovrà produrre un file chiamato `output.txt` che conterrà i seguenti valori:

```
30  
45  
39  
49  
43  
15  
61  
C++ feels like a new language. 61
```

Note:

- Per dimensione di una riga si intende il numero dei caratteri ivi contenuti, escluso il carattere "a capo". Spazi e segni di punteggiatura vanno conteggiati nel computo di ogni riga.
- Si assuma che ogni riga contenga al massimo 255 caratteri.
- Non è concesso fare assunzioni sul numero **massimo** di righe presenti nel file pena l'annullamento dell'esercizio.
- Non è ammesso l'uso di alcuna routine di gestione delle stringhe (ad esempio `strlen` o `strcpy`).
- È consentito l'utilizzo della funzione `getline (char* s, streamsize n)`. Ad esempio, `input.getline(buffer, 10)`.
- Non è consentito l'utilizzo di altre funzioni di libreria "particolari" diverse da quelle specificate sopra o da quelle standard necessarie a risolvere l'esercizio.

- Le uniche assunzioni che si possono fare sull'input e su dimensioni di eventuali strutture/array utilizzate nel file di partenza fornito sono **solo quelle espressamente specificate in questo testo** (e NON quelle riportate nel file fornito, che sono SOLO indicative per consentire di svolgere l'esame).
- All'interno di questo programma **non è ammesso** l'utilizzo di variabili globali o di tipo `static`.
- Si ricorda che, l'esempio di esecuzione è puramente indicativo, e la soluzione proposta NON deve funzionare solo per l'input fornito, ma deve essere robusta a variazioni compatibili con la specifica riportata in questo testo.

Information for graders:

Total of marks: 40