



UNIVERSITÀ
DI TRENTO

Dipartimento di Ingegneria e
Scienza dell'Informazione
DISI - Trento

Programmazione 1

15 - Esercitazione

Giovanna Varni
giovanna.varni@unitn.it

Anno Accademico 2023/2024

Feedback Programmazione I

<https://forms.gle/NNKGmf8YbTnUrSGZ9>

Le risposte inviate attraverso questo form vengono registrate in modo anonimo. Nessuna vostra informazione personale verrà conservata. Il sondaggio è accessibile soltanto se in possesso di credenziali unitn.

Nelle puntate precedenti

- **Stringhe**

```
char stringa [] = "Arancio";
```

- Contiene 8 elementi: i caratteri della stringa e terminatore `'\0'` che viene inserito automaticamente;
- Gli operatori di input e output operano direttamente su stringhe

```
char b[256];  
cin >> b;  
cout << b;
```

Nelle puntate precedenti

- **Funzioni della libreria `<cstring>`**

- `strlen(s)`: ritorna la lunghezza della stringa `s`;
- `strcpy(t, s)`: copia la stringa `s` all'interno della stringa `t` e restituisce `t`;
- `strcmp(t, s)`: restituisce un valore negativo, nullo o positivo se `s` è alfabeticamente minore, uguale o maggiore di `t`;
- ...

Nelle puntate precedenti

- **Argomenti da linea di comando**

```
int main( int argc, char * argv [] )
```

- `argc` indica il numero totale degli elementi della riga di comando (il nome dell'eseguibile + argomenti)
- `argv` è un array multidimensionale che contiene gli argomenti sotto forma di stringhe;

Nelle puntate precedenti

- **Argomenti da linea di comando**

```
./a.out input.txt output.txt 56
```

- `argc = 4`

- `argv: ["/a.out", "input.txt", "output.txt", "56"]`

Per convertire eventuali stringhe in numeri, usare le funzioni `atoi()` o `atof()` della libreria `<cstdlib>`.

Nelle puntate precedenti

- **Stream di I/O e File di Testo**

- Le funzioni per operare su file sono disponibili tramite la libreria `<fstream>`;
- L'utilizzo di uno stream è analogo a `cin` e `cout` (quindi utilizzando gli operatori `>>` e `<<`);
- Ricordarsi sempre di chiudere uno stream una volta aperto;

Nelle puntate precedenti

- **Aprire/Chiudere uno stream**

```
1. fstream input, output;  
2. input.open("file.txt", ios::in);  
3. output.open("output.txt", ios::out);  
  
4. input.close();  
5. output.close();
```


Nelle puntate precedenti

- **Funzioni della libreria `<fstream>`**

- `input.get(c)`: legge un carattere (spazi compresi) dallo stream e lo memorizza nella variabile `c`;
- `input.eof()`: indica se abbiamo raggiunto la fine dello stream;
- `input.fail()`: indica se ci sono stati errori nell'apertura dello stream;
- ...

Nelle puntate precedenti

- **Leggere/Scrivere un file**

```
1. int b;  
2. input>>b;  
3. while(!input.eof() && !input.fail()) {  
4.     output << b << " " ;  
5.     input >> b;  
6. }
```

00 - Punteggio

Si scriva una funzione che riceva come parametro una stringa di caratteri `str` e restituisca come valore di ritorno un numero intero. Dopo avere verificato che la stringa sia composta da caratteri alfabetici (escluso il carattere nullo), la funzione converte eventuali caratteri alfabetici maiuscoli in minuscoli; quindi calcola e restituisce un punteggio per la stringa `str` dato dalla somma delle differenze assolute tra ciascuna coppia di caratteri consecutivi. Ad esempio, data la stringa `str = "aa"`, la funzione restituisce 0; data la stringa `str = "abf"`, la funzione restituisce 5. Se la stringa non contiene solo caratteri alfabetici, la funzione restituisce -1. Si scriva quindi un `main` che, finché l'utente lo desidera, gli chieda di immettere da tastiera una stringa contenente al massimo 31 caratteri, chiami la funzione di cui sopra e, in caso di dato valido, ne stampi a video il valore di ritorno.

01 - Calcolatrice

Scrivere un programma che, dati a linea di comando tre valori, due numeri e un operatore, ritorni a video il risultato dell'operazione matematica definita dall'operatore.

```
./calcolatrice 1 3 +
```

```
Il risultato è 4
```

02 - Impariamo a leggere

Scrivere un programma che, dato in input da linea di comando il nome di un file, ritorni a video il contenuto del suddetto file. Implementare anche dei controlli che avvisino l'utente in caso il numero di argomenti sia sbagliato (manca il file di input) oppure nel caso in cui il file di input non esista.

Documentazione <fstream>:

<http://www.cplusplus.com/reference/fstream/fstream/>

03 - Impariamo a copiare

Scrivere un programma che, dati in input da linea di comando i nomi di due file, A e B, copi il contenuto di A all'interno di B. Implementare anche dei controlli che avvisino l'utente in caso il numero di argomenti sia sbagliato (manca il file di input) oppure nel caso in cui il file di input non esista.

Documentazione <fstream>:

<http://www.cplusplus.com/reference/fstream/fstream/>

04 - Sostituzione di lettera

Scrivere un programma che, dato in input il nome di un file F e una lettera L, stampi a video il contenuto di quel file, sostituendo ogni occorrenza della lettera L con il simbolo “?”

```
"Che magnifica giornata", a
=
"Che m?gnific? giorn?t?"
```

Esercizi Aggiuntivi per Casa

Aggiuntivo - Intersezione di parole

Scrivere un programma che, dati in input due file A e B da riga di comando, generi un terzo file C che contenga tutte le parole presenti in entrambi i file A e B.

E' consentito l'uso delle funzioni `strcmp()` , `strcpy()` della libreria `<cstring>`. Assumete inoltre che i file contengano al massimo 1000 parole e che la lunghezza massima delle singole parole sia 100. **E' consentito aprire al massimo un unico stream per file.**