

Esercitazione 6



UNIVERSITÀ
DEGLI STUDI
DI GENOVA

Dibris

Gualtiero Volpe
gualtiero.volpe@unige.it

Soluzione esercizi quinta esercitazione

Trovate le soluzioni di questi esercizi qui:

https://github.com/gualtierovolpe/fondamenti_di_informatica_2024-25

Le soluzioni delle esercitazioni vengono rilasciate il giorno prima dell'esercitazione successiva.

Un po' di teoria

Si scriva che cosa stamperà in uscita il seguente programma C++, spiegando brevemente il funzionamento della funzione test:

```
#include <iostream>
using namespace std;
```

```
void test(char s[], int& t) {
    int j = 0; t = 0; char c;
    for (int i = 1; s[i] != '\0'; i++) {
        if (s[i] >= 'A' && s[i] <= 'Z') {
            c = s[i];
            s[i] = s[j];
            s[j] = c;
            j++; t++;
        } else {
            c = s[i - 1];
            s[i - 1] = s[i];
            s[i] = c;
        }
    }
}
```

```
int main() {
    char str[] = "beicemDr";

    int n = 0;
    test(str, n);
    cout << str << n << endl;

    str[0] = str[0] - 'A' + 'a';
    test(str, n);
    cout << str << n << endl;

    return 0;
}
```

Un po' di teoria

Prova intermedia del 1 febbraio 2017, esercizio 1. Dato il seguente prototipo di funzione: `int test(double a[], char c, double x, int& r);` e le seguenti definizioni di variabili: `char s = 'v'; int n = 10; double v[3] = {1.5, 2.8, 3.4};` si dica se ciascuna delle seguenti chiamate alla funzione `test` è corretta:

Sì No

- | | | |
|--------------------------|--------------------------|--|
| <input type="checkbox"/> | <input type="checkbox"/> | <code>int res = test(v[], s, 8.6, n);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>double res = test(v, s, 8.6, n);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>int res = test(v, 'z', v[1], n);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>int res = test(v, s, 2.4, 8);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>int res = test(v, n, v[0], n);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>int res = test(v, int(v[2]) + 'a', n, n);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>int res[] = test(v, s, 3.3, n);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>int res = test(v, s, 5.5, &n);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>double res = test(v, s, double(n), n);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> | <code>int res = test(v, s, v[3], n);</code> |

Un po' di teoria

Prova intermedia dell'8 gennaio 2018, esercizio 2. Date le seguenti dichiarazioni di variabili: double x; char str[16]; int a[15]; associare ciascuna delle seguenti istruzioni C++ al suo significato:

```
a[5]++;  
char c = 'a';  
int r = a[2] || a[3];  
cin >> str;  
x = 12.9;  
int test(double v);  
int t = str[0] == str[1];  
x = double(a[6]);  
int y = a[3] + a[4];  
void test(int& k);
```

Operazione di estrazione di una stringa di caratteri
Operazione di assegnamento del valore ad una variabile
Operazione relazionale tra caratteri
Prototipo di funzione con passaggio per riferimento
Conversione esplicita di tipo
Incremento unitario di un numero intero
Operazione logica tra numeri interi
Prototipo di funzione con passaggio per valore
Definizione con inizializzazione di una variabile
Operazione aritmetica tra due numeri interi

Esercizi da Svolgere

Esercizio 6.1 – Generazione di un array di numeri casuali

Si scriva un programma in linguaggio C++ che definisca un array di 30 numeri interi, inizializzi i suoi elementi con valori casuali compresi tra 0 e 99 e, infine, stampi a video l'array così inizializzato.

Nota: per generare un numero intero casuale è possibile utilizzare la funzione `rand()`, che restituisce, in realtà, un numero pseudocasuale tra 0 e `RAND_MAX`. Per far sì che il numero generato da `rand()` sia realmente casuale, occorre inizializzare il «seed» della funzione in maniera casuale. Normalmente, si fa chiamando: `srand(time(NULL))`; `time(NULL)` restituisce i secondi trascorsi dalla mezzanotte del 1 Gennaio 1970. Per poter usare le funzioni `rand` e `time`, si includono gli header `<ctime>` e `<cstdlib>`.

Esercizi da Svolgere

Esercizio 6.2 – Verificare se e quanto due stringhe di caratteri fanno rima

Si scriva in linguaggio C++ la funzione `conta_rima` che riceva come parametri due stringhe di caratteri, conti e restituisca come valore di ritorno il numero di caratteri alfabetici che «fanno rima» (un numero intero), cioè il numero di caratteri alfabetici uguali partendo dal termine delle due stringhe. Ad esempio, se le stringhe passate come parametro sono «mare» e «navigare», la funzione restituisce 3 (ci sono cioè 3 caratteri alfabetici uguali partendo dal termine delle due stringhe). Si scriva quindi un programma che chieda all'utente di immettere da tastiera due stringhe di caratteri, chiami la funzione `conta_rima` e stampi a video il suo valore di ritorno.

Esercizi da Svolgere

Esercizio 6.3 – Verifica dell'appartenenza ad un intervallo

Si scriva una funzione in linguaggio C++ che riceva come parametro un array a di numeri reali, la sua dimensione n e due ulteriori numeri reali x e y . Dopo avere verificato che sia $x < y$, la funzione assegnerà 0 a tutti gli elementi dell'array a che non appartengano all'intervallo $[x, y]$. Inoltre la funzione restituirà come valore di ritorno il numero di assegnamenti effettuati (cioè il numero di elementi non appartenenti all'intervallo $[x, y]$). Se la condizione $x < y$ non è verificata, la funzione restituirà -1. Si scriva quindi un programma per verificare il corretto funzionamento della funzione.

Esercizi da Svolgere

Esercizio 6.4 – Inversione di una stringa

Si scriva in linguaggio C++ la funzione `inverti` che riceva come parametro una stringa `s` di caratteri alfanumerici e la inverta. La funzione, cioè, sovrascrive la stessa stringa in modo tale che risulti letta al contrario. Ad esempio, se la stringa passata come parametro è `s = «informatica»`, la funzione la trasformerà in `s = «acitamrofni»`. La funzione non restituisce alcun valore di ritorno. Si scriva quindi un programma che chieda all'utente di immettere da tastiera una stringa di caratteri, chiami la funzione `inverti` e ne stampi a video il risultato.

Esercizi da Svolgere

Esercizio 6.5 – Conteggio delle occorrenze

Si scriva una funzione in linguaggio C++ che riceva come parametro un array a di numeri interi, la sua dimensione n e un secondo array b di 10 elementi. L'array a contiene numeri interi tra 0 e 9. Come primo passo, la funzione dovrà assegnare ad ogni elemento dell'array b il valore 0. La funzione dovrà quindi calcolare il numero di occorrenze in a di ciascun valore tra 0 e 9 e assegnare il risultato all'elemento corrispondente dell'array b . Ad esempio, l'elemento in posizione 0 dell'array b sarà il numero di occorrenze del valore 0 in a . Se un elemento di a non è un numero tra 0 e 9 la funzione lo ignora. La funzione restituisce come valore di ritorno -1 se almeno un elemento di a non appartiene all'intervallo $[0, 9]$, 0 altrimenti. Si scriva quindi un programma per verificare il corretto funzionamento della funzione.

Esercizi da Svolgere

Esercizio 6.6 – Rimozione delle occorrenze

Si scriva una funzione in linguaggio C++ che riceva come parametro una stringa s e un carattere a . La funzione rimuoverà da s tutte le occorrenze di a . Ad esempio, se $s = \text{"Ottobre"}$ e $a = \text{'t'}$, la stringa s diventerà "Oobre" . La funzione restituirà come valore di ritorno il numero di caratteri eliminati. Si scriva quindi un programma C++ per verificare il corretto funzionamento della funzione.

Esercizi da Svolgere

Esercizio 6.7 – Prova intermedia del 9 gennaio 2017, esercizio 6

Si scriva un programma C++ che dichiari due array a e b di 12 numeri interi, chieda all'utente di inserire da tastiera i valori dell'array a e operi come segue: assegni inizialmente il valore 1 al primo ed all'ultimo elemento dell'array b ; quindi partendo dal penultimo elemento dell'array a e fino al secondo elemento, assegni al corrispondente elemento dell'array b il prodotto dell'elemento precedente e di quello seguente nell'array a . Il programma stamperà infine a video l'array b e ripeterà tutte le operazioni finché l'utente lo desidera. Ad esempio, se $a = \{1, 3, 4, 2, 9, 5, 3, 2, 2, 6, 3, 3\}$, l'array b sarà $b = \{1, 4, 6, 36, 10, 27, 10, 6, 12, 6, 18, 1\}$.

Esercizi da Svolgere

Esercizio 6.8 – Prova intermedia del 5 febbraio 2019, esercizio 6

Un indirizzo IP è costituito da 4 numeri interi che possono assumere valori tra 0 e 255 ed è normalmente rappresentato separando i 4 numeri con il carattere '.'. Ad esempio: 192.168.0.24. Si scriva la funzione C++ strToIP che riceva in ingresso una stringa di caratteri s e un array di quattro numeri interi ip e restituisca come valore di ritorno un numero intero. La stringa s contiene un indirizzo IP formattato come descritto sopra. Dopo avere inizializzato a zero tutti gli elementi dell'array ip, la funzione scandisce la stringa s e assegna a ciascun elemento dell'array ip il corrispondente valore contenuto nella stringa.

Esercizi da Svolgere

Esercizio 6.8 – Prova intermedia del 5 febbraio 2019, esercizio 6 (continua)

Al primo elemento di ip sarà assegnato quindi il primo numero intero che compone l'indirizzo IP rappresentato dalla stringa s, al secondo elemento di ip sarà assegnato il secondo numero intero che compone l'indirizzo IP rappresentato dalla stringa s e così via. Se, ad esempio, si avesse $s = "192.168.0.24"$, si otterrebbe l'array $ip = \{192, 168, 0, 4\}$. Nel caso in cui il valore di almeno uno dei quattro numeri interi componenti l'indirizzo IP non sia valido, la scansione si interrompe e la funzione restituisce -1. Altrimenti la funzione restituisce 1. Per semplicità si assuma che la stringa s sia sempre formattata correttamente.

Esercizi da Svolgere

Esercizio 6.8 – Prova intermedia del 5 febbraio 2019, esercizio 6 (continua)

Si scriva infine un programma C++ che dichiari un array `address` di 4 numeri interi, chieda all'utente di inserire da tastiera una stringa `str` formattata come descritto sopra (la stringa contiene al massimo 15 caratteri), chiami la funzione `strToIP` passandole come parametri `address` e `str` e stampi a video il valore degli elementi dell'array `address` dopo la chiamata della funzione. L'utente può ripetere le operazioni sopra specificate finché lo desidera.

Istruzioni Finali

Cercate di svolgere gli esercizi in autonomia/a piccoli gruppi
(è il miglior modo per vedere se avete capito)

Per ogni domanda, siamo a disposizione

Troverete le soluzioni di questi esercizi qui:

https://github.com/gualtierovolpe/fondamenti_di_informatica_2024-25

Le soluzioni delle esercitazioni vengono rilasciate il giorno prima dell'esercitazione successiva.

