
ESERCITAZIONE 3

ESERCIZIO 1

OBBLIGATORIO

- Scrivere una funzione ricorsiva equivalente alla seguente funzione iterativa. Entrambe le funzioni devono avere gli stessi parametri; non è possibile usare funzioni ausiliarie.

```
int search_a(char s[], int start=0) {  
    int i = start;  
    int tot_a = 0;  
    while (s[i] != '\0' ) {  
        if (s[i] == 'a'){  
            tot_a++;  
        }  
        i++;  
    };  
    return tot_a;  
}
```

ESERCIZIO 2

SONO QUINDI DUE LISTE DIVERSE

NUMERI INTERI

- ▶ Definire le strutture dati necessarie per gestire i risultati di un mondiale di calcio. I risultati sono memorizzati in una lista di partite e una lista di squadre. Per ogni partita, si memorizzano i puntatori alle due squadre e il numero di goal segnati da ognuna. Per ogni squadra si memorizza il nome e i punti ottenuti.
- ▶ Definire opportunamente i tipi di dato, è ammesso l'uso della classe `string`. Definire:
 - ▶ Una funzione che prende come parametro la lista di risultati e ritorna il numero di partite finite in parità
 - ▶ Una funzione che prende come parametro una lista di squadre e ritorna una nuova lista con le sole squadre che hanno ottenuto almeno 10 punti.

LA LISTA È PRESA COME PARAMETRO

ESERCIZIO 3

IL PREZZO E
IL NUMERO DI COPIE HANNO DUE TIPI DI
DATO DIVERSI

- ▶ Definire una classe `Libro` che memorizza titolo, prezzo e numero di copie vendute di un libro. Implementare il costruttore della classe e i seguenti metodi:
 - ▶ Un metodo `vendi_copie(int n)` che incrementa di `n` il contatore di copie vendute;
 - ▶ Un metodo `ricavo()` che restituisce i ricavi ottenuti dalla vendita del libro.
- ▶ Inoltre un libro può essere corredato anche di CD o App. In entrambi i casi è necessario memorizzare anche un codice di attivazione (di tipo intero). Definire le sottoclassi e implementare il relativo costruttore e il metodo necessario per leggere il codice di attivazione.