

Obiettivo

Lettura stringe da input e contare le occorrenze.

Per indicare la fine dell'inserimento dell'input utilizziamo `ctrl + D`

```
#include
#include
#include
#include

using WordFreq = std::map;

// criterio di confronto
struct Greater{
    bool operator()(unsigned long x, unsigned long y) const {
        return x > y;
    }
};

// Conta le stringhe di lunghezza n
struct Shorter{
    bool operator()(const std::string& x, const std::string& y) const {
        return x.size() < y.size();
    }
};

using FreqWord = std::multimap;
// un'alternativa a greater è std::greater

int main(){
    std::cout << "`ctrl + d` per terminare l'inserimento \n";
    std::cout << "Inserire input:\n";

    WordFreq wf;

    std::istream_iterator i(std::cin);
    std::istream_iterator iend;

    // Versione Zaffanella
    for( ; i != iend; ++i){
        const auto& s = *i;
        auto iter = wf.find(s);

        if(iter == wf.end()) // se non trovo niente
            wf.insert(std::make_pair(s, 1));
        else // se trovo qualcosa
            ++iter->second; // oppure ++(*iter).second;
    }

    /** Versione del libro (Stroustrup), che però è poco leggibile
    for( ; i != iend; ++i)
        ++wf[*i];
    */

    // Serve per stampare le occorrenze in ordine decrescente
    FreqWord fw;

    for(const auto& p : wf){
        fw.insert(std::make_pair(p.second, p.first));
    }

    // for(auto iter = wf.begin(); iter != wf.end(); ++iter)
    for(const auto& p : fw){
        std::cout << "La stringa " << p.first
            << " occorre numero " << p.second
            << " volte \n";
    }

    return 0;
}
```