

## Esempio di template di funzione che agisce su vector

I vettori, a differenza degli array, sono passati per valore e non per riferimento, cioè sono ricopiati all'interno della funzione.

Se si vuole un passaggio per riferimento, occorre indicarlo esplicitamente con **&**:

```
template<typename E> void print(vector<E>& v) {
    int n = v.size();
    for(int i=0; i < n; i++) {
        cout << v[i] << endl;
    }
    cout << endl;
}
```

Quest'unica `print` permette di scrivere sullo schermo vettori di elementi di qualsiasi tipo.

13/01/2015

E. Giovannetti - Algoritmica per i Licei con C++ (parte 1)

38

## Esercizio

Definisci un template di funzione

```
template<typename E>
void fillFromInput(E a[], int n) {
```

che riempia l'array `a`, per la lunghezza `n`, di elementi immessi da tastiera.

## Esercizio alternativo

Definisci un template di funzione

```
template<typename E>
void fillFromInput(vector<E> &v)
```

che riempia il vettore `v`, per tutta la sua lunghezza, di elementi immessi da tastiera.

13/01/2015

E. Giovannetti - Algoritmica per i Licei con C++ (parte 1)

39