

INF-103 Pemrograman II

VECTOR & STRING

Dr. Taufik Fuadi Abidin, M.Tech
Irvanizam Zamanhuri, M.Sc

Program Studi Informatika
FMIPA UNIVERSITAS KUALA

<http://www.informatika.unsyiah.ac.id>



Apa itu Vector ?

- Vector adalah sebuah *class template* dalam C++ *Standard Template Library*, yang berfungsi seperti array yang dinamis
- Operator kurung siku [] yang dapat digunakan pada array juga dapat digunakan oleh vector untuk mengakses elemen tertentu
- Elemen vector juga dapat diakses menggunakan fungsi yang diimplementasikan khusus untuk vector
- Penggunaan vector lebih direkomendasikan dibanding array dalam C++



Pendeklarasian Vector

```
vector<type_data> nama_variabel;
```

atau

```
vector<type> nama_variabel(ukuran);
```

Contoh:

```
// vector nilai berukuran 5 bertipe integer  
vector<int> nilai(5);
```

```
// vector ipk berukuran 20 bertipe double  
vector<double> ipk(20);
```



Vector versus Array

Array:

```
tipe nama[ ];
```

atau

```
tipe nama[ukuran];
```

Vector:

```
vector<tipe> nama;
```

```
vector<tipe> nama(ukuran);
```

Salah satu kelebihan dari vector dibanding array adalah kemudahan untuk mengetahui ukuran vector menggunakan fungsi *size()* yang secara khusus telah diimplementasikan dalam *class* vector

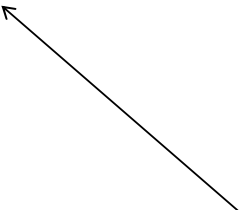
Contoh:

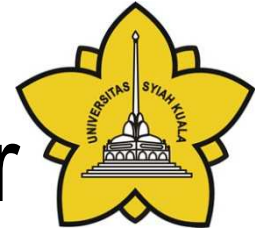


Fungsi *sum* menggunakan Array

```
// n adalah ukuran dari array a dan  
// argumen dari fungsi sum ada 2
```

```
int sum(const int a[], const int n){  
    int s = 0;  
    for (int i = 0; i < n; ++i)  
        s += a[i];  
    return s;  
}
```





Fungsi *sum* menggunakan Vector

```
// argumen fungsi sum hanya 1 dan ukuran  
// diketahui menggunakan fungsi size()
```

```
int sum(const vector<int> a){  
    int s = 0;  
    for (int i = 0; i < a.size(); ++i)  
        s += a[i];  
    return s;  
}
```



Vector dan Operator []

```
#include <iostream>
#include <vector>
#define N 10
using namespace std;

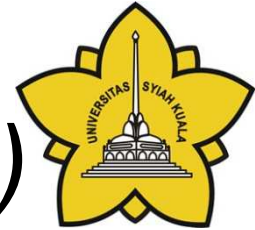
int main(){
    vector<double> nilai(N);

    for (int i = 0; i < N; i++){
        cout << "Nilai " << i+1 << ": ";
        cin >> nilai[i];
    }
    return 0;
}
```



Set Ukuran Vector Saat *run-time*

```
#include <iostream>
#include <vector>
using namespace std;
int main(){
    int jlh_siswa;
    vector<double> nilai;
    cout << "Jumlah siswa: ";
    cin >> jlh_siswa;
    nilai.resize(jlh_siswa);
    for (int i=0; i< (int) nilai.size(); i++){
        cout << "Nilai " << i+1 << ": ";
        cin >> nilai[i];
    }
}
```

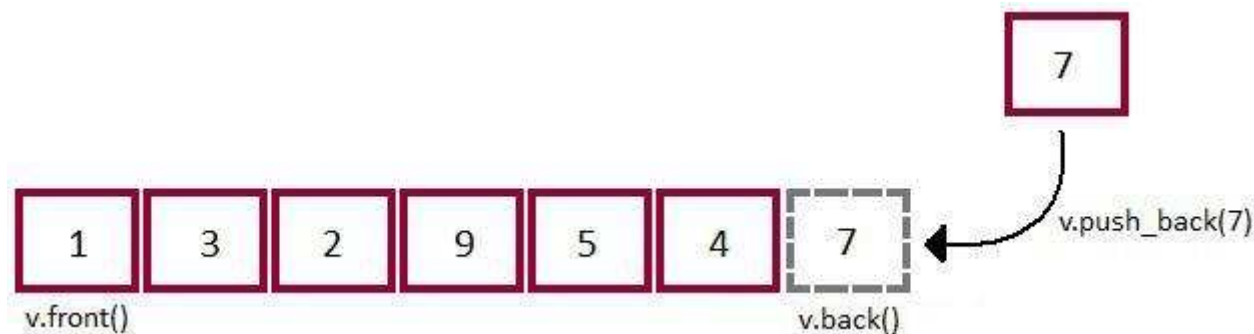
Penggunaan Fungsi *push_back()*

```
#include <iostream>
#include <vector>
using namespace std;
int main(){
    vector<double> nilai;
    int angka;
    char huruf = 'y';
    while (huruf == 'y'){
        cout << "Nilai angka: ";
        cin >> angka;
        nilai.push_back(angka);
        cout << "Input Data Lain (y/n)? ";
        cin >> huruf;
    }
```



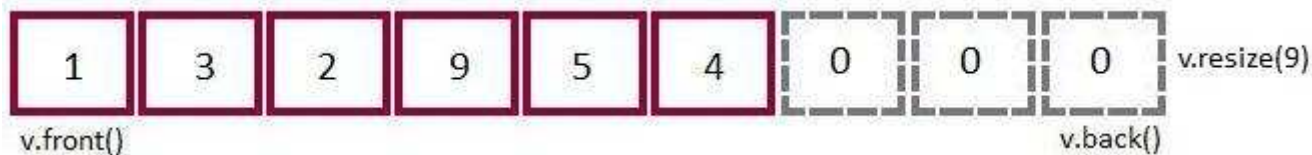
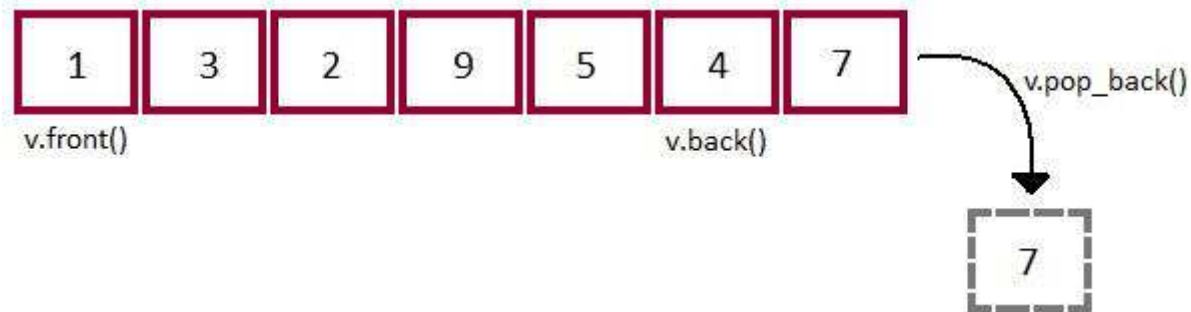
Lanjutan dan visualisasi

```
for (int i=0; i < (int) nilai.size(); i++){  
    cout << "Nilai " <<i+1<< ":\t" << nilai[i];  
    cout << endl;  
}  
return 0;  
}
```





Ilustrasi *pop_back()* dan *resize()*





Penggunaan Iterator

- Iterator merupakan *pointer* dan merupakan cara alternatif untuk mengakses elemen vector
- Iterator merujuk langsung pada elemen vector dan dapat diiterasi sampai akhir elemen vector
- Saat mendeklarasi iterator, tipe dari iterator disesuaikan dengan tipe vector yang akan dirujuk
- Contoh:

```
vector<int>::iterator i;    // i iterator int
```



Iterator: Lanjutan

```
for (int i = 0; i < nilai.size(); i++) {  
    cout << "Nilai " << i + 1 << " : \t";  
    cout << nilai[i] << endl;  
}  
}
```

Dengan iterator:

```
int no = 1;  
vector<double>::iterator i;  
for(i = nilai.begin(); i != nilai.end();  
    ++i, ++no) {  
    cout << "Nilai " << no << " : \t" << *i << endl;  
}  
}
```



String dalam C++

- Bila dalam ANSI/C pendeklarasian *string* dilakukan menggunakan *char **, maka dalam C++ string dapat dideklarasikan langsung menggunakan kata kunci *string* dengan meng-include-kan `<string>`. String merupakan class dalam C++

```
#include <iostream>
#include <string>
using namespace std;
int main () {
    string str = "Hello";
    cout << "str is : " << str << endl;
    return 0;
}
```



String: Fungsi *getline()*

```
#include <iostream>
#include <vector>
#include <string>
using namespace std;
int main(){
    string nama;
    vector<string>vnama;
    do{
        cout << "Nama (Enter = Selesai): ";
        getline (cin, nama);
        if (nama != " "){
            vnama.push_back(nama);
        }
    }while (nama != " ");
```



String: Fungsi *getline()* Lanjutan

// telusuri vector vnama dengan iterator

```
vector<string>::iterator it;
for(it = vnama.begin(); it!=vnama.end();
    ++it){
    cout << "Nama: " << *it << endl;
}
return 0;
}
```




Tugas Bacaan:

C++ Vector

<http://www.cplusplus.com/reference/stl/vector/>

C++ String

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