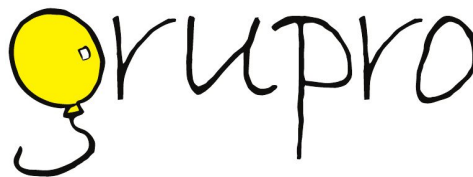




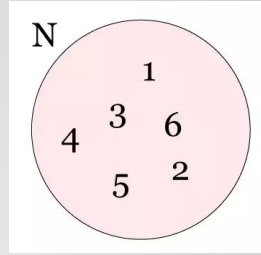
# Curso de Programação Nível Intermediário



**Universidade Federal da Bahia**  
**Instituto de Computação**  
**Departamento de Ciência da Computação**

# **AULA - CONJUNTOS**

# Conjuntos



- Os conjuntos são contêineres que armazenam elementos exclusivos seguindo uma ordem específica (Classificação).
- Cada elemento deve ser um valor único.
- O valor dos elementos em um conjunto não pode ser modificado uma vez inseridos no contêiner (os elementos são sempre constantes),
  - mas podem ser inseridos ou removidos do conjunto.
- Geralmente, a complexidade de tempo de operações como inserção e exclusão no conjunto em C++ é  $O(\log n)$

# Vetores - set

```
#include <iostream>
```

```
#include <set>
```

```
using namespace std; // set -> std
```

```
int main() {
```

```
    set<int> c;
```

```
    for(int i=0; i < 5; i++) {
```

```
        c.insert(i*5);
```

```
    }
```

```
    set<int>::iterator it;
```

```
    for(it = c.begin(); it != c.end(); it++){
```

```
        cout << *it << endl;
```

```
    }
```

```
}
```

# Vetores - set

**#include <set>**

Biblioteca necessária para o uso de conjuntos.

**set<TIPO> S;**

TIPO é qualquer tipo da linguagem: int, char, double, vector<float>, ...

**S.clear();**

Remove todos os elementos do conjunto.

**S.insert(ELEMENTO);**

Adiciona ELEMENTO do tipo TIPO no conjunto, **caso ele ainda não exista!**

**S.begin();**

Retorna iterator para o elemento inicial do conjunto.

**S.end();**

Retorna iterator para o fim do conjunto, **e não contém um elemento!**

**S.erase(ELEMENTO);**

Remove ELEMENTO, caso ele existe no conjunto.

**S.erase(it);**

Remove elemento apontado pelo iterator **it**.

**S.find(ELEMENTO);**

Se ELEMENTO existe, retorna um iterator para ele, caso contrário retorna **S.end()**.

**S.count(ELEMENTO);**

Se ELEMENTO existe, retorna um 1, caso contrário retorna 0.

**S.size();**

Retorna a quantidade de elementos do conjunto.

# Exemplo - set

```
#include <iostream>
#include <set>
#include <iterator>
using namespace std;
int main() {
    int v[]={ 12,12,37,64,15,40,5,3 };
    set<int> S(v,v+5);    // c1 recebe os 5 primeiros valores exclusivos de v
    set<int>::iterator it;
    for(it = S.begin(); it != S.end(); it++)
        cout << *it << " ";
    cout << endl;
    cout << "Existe o elemento 12 : " << S.count(12) << endl;
    it = S.find (37);
    it = prev(it); // Retorna um iterator para o elemento anterior de it.
    cout << "Elemento anterior de 37 :" << *it << endl;
    S.erase(it);
    S.erase(12);
    it = next(S.begin()); //Retorna um iterator para o elemento posterior de S.begin()
    cout << "Segundo elemento:" << *it << endl;
}
```

## Saída:

**\$ g++ ex.cpp -std=c++11**

**\$ a.out**

12 15 37 64

Existe o elemento 12 : 1

Elemento anterior de 37 : 15

Segundo elemento : 64

# Vetores - set

Saiba mais em:

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