Iteradores

Carlos E. Alvarez[1].

[1] Dep. de Matemáticas aplicadas y Ciencias de la Computación, Universidad del Rosario

2019-II

Apuntador como iterador

```
int main(){
  const int size = 10;
  int myarray[size];
  for(int i = 0; i < size; i++)
    myarray[i] = 12-i;

  int *it = nullptr;

  for(it = myarray; it != myarray+size; it++)
    std::cout << *it << " ";
  std::cout << "\n";

  return 0;
}
```

Iteradores: límites

Chequeo de límites

```
int main(){
  const int size = 4;
  std::vector<int> v(size, 1);
  std::vector<int>::iterator it; //iterator of vector

  std::cout << "Without iterators = ";
  for(int i = 0; i < size; i++)
    std::cout << v[i] << " ";

  std::cout << "\nWith iterators = ";
  for(it = v.begin(); it != v.end(); it++)
    std::cout << *it << " ";
```

Iteradores: límites

```
  std::cout << "\nRemoving element...";
  v.pop_back(); //Remove an element

  std::cout << "\nWithout iterators = ";
  for(int i = 0; i < size; i++)
    std::cout << v[i] << " ";

  std::cout << "\nWith iterators = ";
  for(it = v.begin(); it != v.end(); it++)
    std::cout << *it << " ";
  std::cout << "\n";

  return 0;
}
```

Iteradores: cambiar longitud durante un ciclo

```cpp
int main(){
  std::vector<int> v = {1,2,3};
  std::vector<int>::iterator it;

  //Insertar un elemento durante un ciclo
  for(it = v.begin(); it != v.end(); it++){
    if(it == v.begin()+1){
      it = v.insert(it, 5);
    }
  }

  for(int i = 0; i < v.size(); i++)
    std::cout << v[i] << " ";
  std::cout << "\n";

  return 0;
}
```