

Exemples d'utilització de la STL

Els següents exemples es suministren juntament en la implementació original de HP de les STL. Seguir el seu codi amb la sortida a pantalla corresponent us pot ajudar a entendre millor com funcionen. Per provar els exemples en el Visual hauríeu de canviar la part dels includes.

- Els fitxers de la STL del Visual no porten extensió .h.
- Han canviat 'algo.h' per **algorithm**. A més, algunes funcions que en la implementació original estaven a 'algo.h' ara no estan a **algorithm** (per exemple, la funció accumulate està a **numeric**)
- Hi ha un iostream especial per quan es treballa amb les STL: **iostream** (sense extensió)
- A continuació dels includes dels fitxers de la STL, cal posar **using namespace std;**

Search

```
#include <algo.h>
#include <iostream.h>
#include <vector.h>

int main ()
{
    typedef vector <int> IntVec;
    IntVec v1 (10);
    iota (v1.begin (), v1.end (), 0);
    IntVec v2 (3);
    iota (v2.begin (), v2.end (), 50);
    ostream_iterator <int> iter (cout, " ");

    cout << "v1: ";
    copy (v1.begin (), v1.end (), iter);
    cout << endl;
    cout << "v2: ";
    copy (v2.begin (), v2.end (), iter);
    cout << endl;

    IntVec::iterator location;
    location = search (v1.begin (), v1.end (), v2.begin (), v2.end ());

    if (location == v1.end ())
        cout << "v2 not contained in v1" << endl;
    else
        cout << "Found v2 in v1 at offset: " << location - v1.begin () << endl;

    iota (v2.begin (), v2.end (), 4);
    cout << "v1: ";
    copy (v1.begin (), v1.end (), iter);
    cout << endl;
    cout << "v2: ";
    copy (v2.begin (), v2.end (), iter);
    cout << endl;

    location = search (v1.begin (), v1.end (), v2.begin (), v2.end ());

    if (location == v1.end ())
        cout << "v2 not contained in v1" << endl;
    else
        cout << "Found v2 in v1 at offset: " << location - v1.begin () << endl;

    return 0;
}
```

Sortida a Pantalla

```
v1:  0 1 2 3 4 5 6 7 8 9
v2 : 50 51 52
v2 not contained in v1
v1:  0 1 2 3 4 5 6 7 8 9
v2 :  4 5 6
Found v2 in v1 at offset: 4
```

Count

```
#include <algo.h>
#include <iostream.h>
#include <vector.h>

int main ()
{
    vector <int> numbers(100);
    for (int i = 0; i < 100; i++)
        numbers[i] = i % 3;
    int elements = 0;
    count (numbers.begin (), numbers.end (), 2, elements);
    cout << "Found " << elements << " 2's." << endl;
    return 0;
}
```

Sortida a Pantalla

Found 33 2's

Deque

```
#include <iostream.h>
#include <deque.h>
#include <algo.h>

int main ()
{
    deque d;
    d.push_back (4);
    d.push_back (9);
    d.push_back (16);
    d.push_front (1);
    for (int i = 0; i < d.size (); i++)
        cout << "d[" << i << "] = " << d[i] << endl;
    cout << endl;
    d.pop_front (4);
    d[2] = 25;
    for (i = 0; i < d.size (); i++)
        cout << "d[" << i << "] = " << d[i] << endl;
    return 0;
}
```

Sortida a Pantalla

```
d[0] = 1
d[1] = 4
d[2] = 9
d[3] = 16
```

```
d[0] = 4
d[1] = 9
d[2] = 25
```

Rotate

```
#include <algo.h>
#include <iostream.h>
#include <vector.h>

int main ()
{
    vector <int> v1 (10);
    iota (v1.begin (), v1.end (), 0);
    ostream_iterator <int> iter (cout, " ");
    copy (v1.begin (), v1.end (), iter);
    cout << endl;
    for (int i = 0; i < v1.size (); i++)
    {
        rotate (v1.begin (), v1.begin (), v1.end ());
        ostream_iterator <int> iter (cout, " ");
        copy (v1.begin (), v1.end (), iter);
        cout << endl;
    }
    cout << endl;
    return 0;
}
```

Sortida a Pantalla

```
0123456789
0123456789
1234567890
3456789012
6789012345
0123456789
5678901234
1234567890
8901234567
6789012345
5678901234
```

Includes

```
#include <algo.h>
#include <iostream.h>
#include <string.h>
#include <vector.h>

bool compare_strings (const char* s1_, const char* s2_)
{
    return ::strcmp (s1_, s2_) < 0 ? 1 : 0;
}

char* names[] = { "Todd", "Mike", "Graham", "Jack", "Brett"};

int main ()
{
    const unsigned nameSize = sizeof (names)/sizeof (names[0]);
    vector <char*> v1(nameSize);
    for (int i = 0; i < v1.size (); i++)
    {
        v1[i] = names[i];
    }
    vector <char*> v2 (2);
    v2[0] = "foo";
    v2[1] = "bar";
    sort (v1.begin (), v1.end (), compare_strings);
    sort (v2.begin (), v2.end (), compare_strings);

    bool inc = includes (v1.begin (), v1.end (),
                        v2.begin (), v2.end (),
                        compare_strings);

    if (inc)
        cout << "v1 includes v2" << endl;
    else
        cout << "v1 does not include v2" << endl;
    v2[0] = "Brett";
    v2[1] = "Todd";
    inc = includes (v1.begin (), v1.end (),
                  v2.begin (), v2.end (),
                  compare_strings);

    if (inc)
        cout << "v1 includes v2" << endl;
    else
        cout << "v1 does not include v2" << endl;
    return 0;
}
```

Sortida a Pantalla

```
v1 does not include v2
v1 includes v2
```