Subject: Vector of Vector
Posted by forlano on Wed, 25 Oct 2006 20:09:47 GMT
View Forum Message <> Reply to Message

Hi,

I needed to substitute some structure build in C with pointers and I've recognised that it is equivalent to a container of Vector of Vector of the beautifull NTL library (U++ Core). I have not seen any example on this specific structure on the tutorials, so for me was not evident the syntax. After many trials ed errors the following code run with success. It populated the container, print it and produce the sort of a vector. What is very nice, for me, is that the structure became a matrix the which element can be accessed by index without stupid iterators. Marvelous!
I do not know if there are other more direct way to achieve the same purpose. If yes I'm interested in it. Perhaps this topic can be matter of a new tutorial on the NTL that merit to be known more in deep.

Luigi

```
#include <Core/Core.h>

CONSOLE_APP_MAIN
{  FileOut out( "C:\\upp\\out\\MSC8\\out.txt"); //set your dir and file name

   Vector<  Vector<int>  > array;  // vector of vector
   Vector<int> vec;

 //create a vector and add it to the array
 vec.Add(80);
 vec.Add(20);
 vec.Add(30);
 vec.Add(50);
 vec.Add(90);
   array.Add(vec);

 //create another vector and add it to the array
       vec.Clear();
 vec.Add(100);
 vec.Add(200);
 vec.Add(300);
   array.Add(vec);

 //create a vector and add it to the array
       vec.Clear();
 vec.Add(1000);
 vec.Add(2000);
 vec.Add(3000);
 vec.Add(3000);
```

```
    array.Add(vec);

    for(int i = 0; i < array.GetCount(); i++)
   {
     for(int j = 0; j < array[i].GetCount(); j++) out.Put( AsString(array[i][j]) + ", ");
     out.Put("\n");
   }
     out.Put("\n\n\n");

 // sort the first vector of the array
 Sort(array[0]);
    for(int i = 0; i < array.GetCount(); i++)
   {
     for(int j = 0; j < array[i].GetCount(); j++) out.Put( AsString(array[i][j]) + ", ");
     out.Put("\n");
   }

 out.Close();
}
```

Subject: Re: Vector of Vector
Posted by mirek on Wed, 25 Oct 2006 23:35:31 GMT
View Forum Message <> Reply to Message

This is not very effective way to do it (in fact, it is as slow as STL . Much better is to create it "in-place":

```
Vector< Vector<int> > x;
Vector<int>& n = x.Add();
n.Add(1);
....
```

Alternatively, you can use AddPick - that one would destroy source Vector avoiding the copy.

Mirek

Subject: Re: Vector of Vector
Posted by mirek on Wed, 25 Oct 2006 23:36:27 GMT
View Forum Message <> Reply to Message

P.S.: Investigate "LOG" and "DUMP"..

Subject: Re: Vector of Vector
Posted by forlano on Thu, 26 Oct 2006 07:46:23 GMT
View Forum Message <> Reply to Message

luzr wrote on Thu, 26 October 2006 01:36P.S.: Investigate "LOG" and "DUMP"..

If LOG() is intented to print on a file in the directory of the application, then does not work with me. I'll try DUMP.
Thanks for the improvement of the code... but I need some time to understand this syntax .

Luigi