
Subject: Passing Vectors as arguments of functions.
Posted by [281264](#) on Thu, 26 Aug 2010 16:54:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

A simple query, can it be a Vector (or any other template) passed as an argument of a function?
Are there some limitations?

Cheers,

Javier.

Subject: Re: Passing Vectors as arguments of functions.
Posted by [Oblivion](#) on Thu, 26 Aug 2010 17:12:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Yes you can use vectors (or other templates) as arguments, and it is very simple. If you already have the UPP package installed, just check out the bazaar/rss package. It uses vectors and vectormaps as arguments

Regards.

Subject: Re: Passing Vectors as arguments of functions.
Posted by [281264](#) on Thu, 26 Aug 2010 18:29:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Rss code is not available in bazaar (the link does not work).

Javier

Subject: Re: Passing Vectors as arguments of functions.
Posted by [281264](#) on Thu, 26 Aug 2010 18:43:07 GMT
[View Forum Message](#) <> [Reply to Message](#)

for example, in this simple code:

```
#include <iostream>
#include <CtrlLib/CtrlLib.h>
using namespace std;
using namespace Upp;
```

```

struct A:Moveable<A>{

    int x,y,z;
    A(int x1, int y1, int z1){x=x1;y=y1;z=z1;}

};

void multi (Vector<A> a, Vector<A> b, Vector<A> c);

int main(){

    Vector<A> vector_A1,vector_A2,vector_A3;

    vector_A1.Add(A(10,20,30));
    cout<<vector_A1[0].x<<"\n";
    cout<<vector_A1[0].y<<"\n";
    cout<<vector_A1[0].z<<"\n";

    vector_A2.Add(A(40,50,60));

    multi(vector_A1,vector_A2,vector_A3);
    //cout<<vector_A3[0].x<<"\n";
    //cout<<vector_A3[0].y<<"\n";
    //cout<<vector_A3[0].z<<"\n";

    return 0;
}

void multi (Vector<A> a, Vector<A> b, Vector<A> c){

    c.Add(A(a[0].x*b[0].x,a[0].y*b[0].y,a[0].z*b[0].z));

}

```

why values are not passed to vector_A3? is it correct the way of passing the Vectors?

Cheers,

Javier

Subject: Re: Passing Vectors as arguments of functions.
 Posted by [281264](#) on Thu, 26 Aug 2010 18:45:15 GMT

[View Forum Message](#) <> [Reply to Message](#)

I realise that vector_A3 has to be passed by reference.

Subject: Re: Passing Vectors as arguments of functions.

Posted by [Oblivion](#) on Thu, 26 Aug 2010 19:01:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

A simple example code (nevermind its absurdity):

```
Vector<int> index;  
VectorMap<String, String> person;
```

```
void Foo::AddPersons(Vector<int>& i, VectorMap<String, String>& p)  
{  
    bla bla...  
}
```

// In your application, all you have to do is to call the function/method properly.

```
Foo::AddPersons(index, person);
```

And here is some information about containers on UPP

[http://www.ultimatepp.org/srcdoc\\$Core\\$Tutorial\\$en-us.html](http://www.ultimatepp.org/srcdoc$Core$Tutorial$en-us.html)

[http://www.ultimatepp.org/srcdoc\\$Core\\$Caveats\\$en-us.html](http://www.ultimatepp.org/srcdoc$Core$Caveats$en-us.html)

I hope this will help you.

And FYI, Rss example is in the UPP package/bazaar section. Not on the web page.

Regards.

Subject: Re: Passing Vectors as arguments of functions.

Posted by [281264](#) on Thu, 26 Aug 2010 20:39:49 GMT

[View Forum Message](#) <> [Reply to Message](#)

Roger!

Subject: Re: Passing Vectors as arguments of functions.
Posted by [cbpporter](#) on Thu, 26 Aug 2010 21:18:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

You should always pass containers as a reference, otherwise you risk losing the content of your original container.
