
Subject: Small fix for Find in Algo.h

Posted by [gridem](#) on Wed, 11 Mar 2009 17:38:04 GMT

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

I think that this patch solves the problem for Find function (it should compare the values):

```
template <class T, class V>
T Find(T ptr, T end, const V& value)
{
- return Find(ptr, end, value, StdEqual<T>());
+ return Find(ptr, end, value, StdEqual<V>());
}
```

Subject: Re: Small fix for Find in Algo.h

Posted by [mirek](#) on Wed, 11 Mar 2009 20:15:30 GMT

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

Thanks!

Mirek

Subject: BUG for Find in Algo.h

Posted by [ratah](#) on Mon, 12 Dec 2011 10:14:39 GMT

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

Hello,

I have a little bug when i use find algorithm. Here is my test :

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

```
using namespace Upp;
```

```
class CAppli:Moveable<CAppli>
```

```
{
public:
```

```
    String m_appli;
    int m_count;
```

```
    CAppli(const String& appli)
```

```
{
    m_appli = appli; m_count = 0;
}
```

```

String ToString()
{
    String mycout;

    mycout << " APPLI : " << m_appli;

    return mycout;
}

void update()
{
    m_count++;
}

};

bool operator==(const CAppli& oAppli, const CAppli& oAppliTest)
{
    Cout() << "\n    -> test : " << oAppliTest.m_appli << " =? " << oAppli.m_appli;

    if( oAppli.m_appli == oAppliTest.m_appli)
    {
        Cout() << "\n    -> result : " << oAppliTest.m_appli << " is equal to " << oAppli.m_appli;
        return true;
    }
    else
    {
        Cout() << "\n    -> result : " << oAppliTest.m_appli << " is different of " << oAppli.m_appli;
        return false;
    }
}

class CJSON: public Json
{
public :

    Vector<CAppli> vAppli;

    CJSON operator<< (const CAppli& oAppli)
    {
        Cout() << "\nTry to add " << oAppli.m_appli;

        Vector<CAppli>::Iterator it;
        it = Find(vAppli.Begin(), vAppli.End(), oAppli);

        if(it != vAppli.End())
        {
            //CAppli apptmp = *it;           //<---- IT'S BUG !!!!!!!

```

```

    //apptmp.update();
    //Cout() << "\n  <- Why I fall here even if " << apptmp.m_appli << " is different of " <<
oAppli.m_appli;

    Cout() << "\n  <- Why I fall here?";
}
else
{
    Cout() << " not yet in the vector. ";

    vAppli.Add(oAppli);
    Cout() << oAppli.m_appli << " added";

}

    Cout() << "\n-----\n\n";

    return (*this);
}
};

```

```

CONSOLE_APP_MAIN
{
    CAppli appl1("APP1"), appl2("APP2"), appl3("APP3");

    CJSON ojson;
    ojson << appl1 << appl2 << appl3 << appl2;
}

```

Thanks for your analyse and reply

Best regards,

Ratah

Subject: Re: BUG for Find in Algo.h
 Posted by [mirek](#) on Tue, 13 Dec 2011 16:16:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

U++ iterators are allowed to be assigned NULL, so it makes more sense to return NULL in 'fail' case by U++ algos.

It is written somewhere in doc, but perhaps should have been emphasized more...

So:

```

if(it)
{
    //CAppli apptmp = *it;          //<---- IT'S BUG !!!!!!!
    //apptmp.update();
    //Cout() << "\n <- Why I fall here even if " << apptmp.m_appli << " is different of " <<
oAppli.m_appli;

    Cout() << "\n <- Why I fall here?";
}

```

Notes:

U++ prefers index notation, so FindIndex(...) < 0 would be more 'U++ way'.

And based on the name of class, U++ now has pretty good JSON support:

[http://www.ultimatepp.org/reference\\$JSON\\$en-us.html](http://www.ultimatepp.org/reference$JSON$en-us.html)

Mirek

Subject: Re: BUG for Find in Algo.h
 Posted by [ratah](#) on Mon, 19 Dec 2011 08:26:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks a lot

Subject: Re: BUG for Find in Algo.h
 Posted by [gridem](#) on Mon, 19 Dec 2011 09:10:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

There are 2 issues:

1. You should check iterator for NULL. This is difference between STL and U++.
2. You should use reference instead of value:

The correct code is:

```

if (it != NULL)
{

```

```
CAppli& apptmp = *it;           // use reference  
apptmp.update();
```
