

---

Subject: Pass data of Vector<Vector<double>> into a function call

Posted by [sinpeople](#) on Wed, 17 Mar 2021 07:13:04 GMT

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

---

Hi folks,

I need to pass a data variable of type Vector<Vector<double>> into a function call. What's the best way to do it?

Inside the function call, a clone will be created according to the data passed in, without modifying the value passed in.

Any example of general guidelines on doing this? Thank you!

Best Regards

David WANG

---

---

Subject: Re: Pass data of Vector<Vector<double>> into a function call

Posted by [Didier](#) on Wed, 17 Mar 2021 20:53:31 GMT

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

---

Well quite simple question:

just put:

```
void function( const Vector<Vector<double>>& var);
```

---

---

Subject: Re: Pass data of Vector<Vector<double>> into a function call

Posted by [Xemuth](#) on Wed, 17 Mar 2021 22:57:15 GMT

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

---

Hello,

if you want to clone your vector, you can simply use clone :

This exemple show a function taking a vector<vector<double>> in entry cloning it, add 1 to all of double, then return it by moving the clone.

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

```
using namespace Upp;
```

```
Vector<Vector<double>> PrintVector(const Vector<Vector<double>>& vector);
```

```
CONSOLE_APP_MAIN
```

```
{
```

---

```

//original vector of vector of double
Vector<Vector<double>> original{{2.0,3.0},{4.0,5.0}};

//pick allow you to move a value between variable without having to do copy
Vector<Vector<double>> ret = pick(PrintVector(original));

//We print the vector and all is double
for(const Vector<double>& vec : ret){
    for(const double& d : vec){
        Cout() << d;
    }
    Cout() << "\n";
}
}

Vector<Vector<double>> PrintVector(const Vector<Vector<double>>& vector){
//Clone allow you to clone a variable (the class you want to copy must have a copy constructor)
Vector<Vector<double>> myClone = clone(vector);

//We add + 1 to all double of our clone
for(Vector<double>& vec : myClone){
    for(double& d : vec){
        d += 1.0;
    }
}

//we return our clone not by copying but by moving it
return pick(myClone);
}

```

---

Subject: Re: Pass data of Vector<Vector<double>> into a function call  
 Posted by [mirek](#) on Fri, 19 Mar 2021 12:26:36 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Sometimes it is good to use r-value reference, as it gives you freedom in usage:

```

Vector<Vector<double>> AdjustVector(Vector<Vector<double>>&& vector){
    for(Vector<double>& vec : vector){
        for(double& d : vec) {
            d += 1.0;
        }
    }
    return pick(vector);
}

```

```
CONSOLE_APP_MAIN
{
Vector<Vector<double>> x { { 1, 2 }, { 3, 4 } };

auto y = AdjustVector(clone(x));

DDUMP(x);
DDUMP(y);

y = AdjustVector(pick(x));

DDUMP(x);
DDUMP(y);
}
```

Mirek

---