

Forum:[Hack Forums](#)

Thread: [Question Inherit the functions of a metaprogrammed class](#)

minerw - October 5, 2020, 11:45 am

Hi, while trying to hone my skills by creating a meta-programmed class construct along with the concepts of inheritance I came across this oddity (it's a simplified version of what I'm trying to do):

C++:

```
#include <iostream>
```

```
template<class T>
class BASE
{
public:
    void func()
    {
        std::cout << "Hello world";
    }
};
```

```
template<class T>
class DERIVED : public BASE<T>
{
public:

    void func2()
    {
        func(); // <-- 'func': identificatore non trovato
    }
};
```

```
int main()
{
    DERIVED<int> obj;
    obj.func2();
}
```

As you can guess from the comment, the func (); included in the scope of func2 (); is not recognized, while in the case of a non meta-programmed class the code would be compiled without errors:

C++:

```
#include <iostream>
```

```
class BASE
{
public:
    void func()
```

```

    {
        std::cout << "Hello world";
    }

};

class DERIVED : public BASE
{
public:

    void func2()
    {
        func();
    }
};

int main()
{
    DERIVED obj;
    obj.func2();
}

```

I've also tried making func2 (); a virtualization of func () ;, but the function call func (); gives rise to a buffer overflow.

```

C++:
#include <iostream>

template<class T>
class BASE
{
public:
    virtual void func()
    {
        std::cout << "Hello world";
    }

};

template<class T>
class DERIVED : public BASE<T>
{
public:

    void func() override
    { // Buffer overflow
        func();
    }
};

int main()
{
    DERIVED<int> obj;
}

```

```
    obj.func();  
}
```

So I wonder, how is it possible to do such a thing: ???:?

hell-o-world - *October 7, 2020, 5:44 am*

Would love to see this question answer ? I started learning class and struct yesterday

Thanks have a good day all !