

---

Subject: method passed as parameter in function  
Posted by [BetoValle](#) on Sun, 03 Oct 2021 13:04:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

These methods below within the class do not allow it to be compiled this way! is it possible something analogous so that the fx\_ method inside the class compile on u++? If positive how would it be?

Thanks!

```
class ab {
public:
    String fx1( Value v1){

        return "fx1";
    }
    String fx2( Value v2){

        return "fx2";
    }
    String fx3( Value v3){

        return "fx3";
    }
    String fx_( Value v , String (ab::*func)(Value) ){

        return func(v);

    }

};

CONSOLE_APP_MAIN
{

    Value v=1;
    ab f
    String x= f.fx_( v, ab::&fx1);
    Cout() << x << EOL;
}
```

---

---

Subject: Re: method passed as parameter in function  
Posted by [pvictor](#) on Mon, 04 Oct 2021 11:15:21 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

BetoValle wrote on Sun, 03 October 2021 18:04Hi,

These methods below within the class do not allow it to be compiled this way! is it possible something analogous so that the fx\_ method inside the class compile on u++? If positive how would it be?

Thanks!

```
class ab {
public:
String fx1( Value v1){

return "fx1";
}
String fx2( Value v2){

return "fx2";
}
String fx3( Value v3){

return "fx3";
}
String fx_( Value v , String (ab::*func)(Value) ){

return func(v);

}

};

CONSOLE_APP_MAIN
{

Value v=1;
ab f
String x= f.fx_( v, ab::&fx1);
Cout() << x << EOL;
}
```

Try this:

```
#include <Core/Core.h>
using namespace Upp;

class ab {
public:
String fx1( Value v1){
return "fx1";
}
String fx2( Value v2){
return "fx2";
}
String fx3( Value v3){
return "fx3";
}
String fx_( Value v , String (ab::*func)(Value) ){
return (this->*func)(v);
}
};
```

```
CONSOLE_APP_MAIN
{
Value v=1;
ab f;
String x= f.fx_( v, &ab::fx1);
Cout() << x << EOL;
}
```

---

Subject: Re: method passed as parameter in function  
Posted by [BetoValle](#) on Mon, 04 Oct 2021 11:46:06 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

now this work fine! Thanks!

---