```
 1: #include <algorithm>
 2: #include <iostream>
 3: #include <cmath>
 4: using namespace std;
 5:
 6: class A{
 7:     public:
 8:     A(double x): _x(x), _n(1) {};
 9:
10:     A(A const & rhs) = default;
11:     A(A       && rhs) = default;
12:     A& operator=(A const & rhs) = delete;      // not available with const member
13:     A& operator=(A       && rhs) = default;
14:     ~A()                         = default;
15:
16:     double pow() const
17:     {
18:         return std::pow(_x,_n);
19:     }
20:
21:     void SetPower(int n)
22:     {
23:         const_cast<int&>(_n) = n;              // trick
24:     }
25:
26:     private:
27:     double _x;
28:     const int    _n;
29: };
30:
31: int main()
32: {
33:     //--------------------  change const member   --------------
34:     A xa(1.2345);
35:
36:     xa.SetPower(3);
37:     cout << xa.pow() << endl;
38:
39:     // ----------
40:     A xb(xa);
41:     A xc(xa);
```

```
42:     xb.SetPower(4);
43:     //xc = xb;    // not available with const member
44:
45:
46:     return 0;
47: }
48:
```