

Programmation Orienté Objet

LIV 2022-2023
Travaux Dirigés

Site du cours : <https://defelice.up8.site/poo.html>

`g++ mon_fichier.cpp -std=c++17 -Wall -Wextra -fsanitize=address -o mon_programme`

Les exercices marqués d'un @ sont à faire dans un second temps.

Remarque :

- Ne pas s'aider de machine sauf pour confirmer.
- Les morceaux de codes ne correspondent pas à des situations pratiques, ils sont là pour illustrer les mécanismes du C++.
- La mise en page du code laisse à désirer son but est d'économiser de la place.
- Il n'y a aucune erreur de syntaxe pure : pas de parenthèse/accolade/point virgule manquant(e) etc ... (ou plutôt : s'il y en a elles ne sont pas intentionnelles)

1 Membre MI

Exercice 1. *membres MI1.2*

Qu'affiche l'exécution de ce code ?

```
1 #include<iostream>
2 struct A
3 {
4     int a;
5     void methode(void)
6     {
7         a=1;
8     }
9     void setMet(int a)
10    {
11        a=5;
12    }
13 };
14 int main()
15 {
16     A obj;
17     obj.a=0;
18     std::cout << obj.a << std::endl;
19     obj.methode();
20     std::cout << obj.a << std::endl;
21     obj.setMet(5);
22     std::cout << obj.a << std::endl; // piege
23     return 0;
24 }
```

Exercice 2. *membres MI2*

Pouquoi ce code ne compile pas ?

```
1 #include<iostream>
2 struct A
3 {
```

```

4      int a;
5      void methode(void)
6      {
7          a=1;
8      }
9      void methode3(void)
10     {
11         methode();
12     }
13 };
14 int main()
15 {
16     A obj;
17     obj.methode3();
18     methode();
19     std::cout << obj.a << std::endl;
20     return 0;
21 }

```

Exercice 3. *membres MI5*

Qu'affiche ce code ?

```

1 #include<iostream>
2 void methode1(void) { std::cout << "1F"<<std::endl; }
3 struct A
4 {
5     int a;
6     void methode1(void) { std::cout << "1M"<<std::endl; }
7     void methode2(void) { std::cout << "2M"<<std::endl; }
8     void methode3(void) {
9         this->methode2();
10        methode1();
11    }
12 };
13 int main()
14 {
15     A obj;
16     obj.methode3();
17     return 0;
18 }

```

Exercice 4. *membres MI3*

Qu'affiche ce code ?

```

1 #include<iostream>
2 void methode1(void) { std::cout << "1F"<<std::endl; }
3 struct A
4 {
5     int a;
6     void methode1(void) { std::cout << "1M"<<std::endl; }
7     void methode2(void) { std::cout << "2M"<<std::endl; }
8     void methode3(void) {
9         void methode1(void);
10        this->methode2();

```

```

11     methode1 ();
12 }
13 };
14 int main ()
15 {
16     A obj;
17     obj.methode3 ();
18     return 0;
19 }

```

2 Pub/Priv PP

Exercice 5. *publique/privé PPI1*

Ce code compile-t-il ? si non pourquoi ?

```

1 struct A{
2     private :
3         int atr ;
4 };
5 int main(void){
6     A ob;
7     ob.atr=1;
8     return 0;
9 }

```

Exercice 6. *publique/privé PPI3*

Ce code compile-t-il ? si non pourquoi ?

```

1 class A{
2     public :
3         int atr ;
4 };
5 int main(void)
6 {
7     A ob;
8     ob.atr=1;
9     return 0;
10 }

```

Exercice 7. *publique/privé PPI5*

Ce code compile-t-il ? si non pourquoi ?

```

1 class A{
2     private :
3         int atr ;
4     public :
5         int somme(A obj)
6         {
7             return obj.atr+this->atr ;
8         }
9 };
10 int main(void){
11     return 0;

```

12 | }

Exercice 8. *publique/privé PPI4*

Ce code compile-t-il ? si non pourquoi ?

```
1 class A{
2     int atr;
3 };
4 int main(void){
5     A ob;
6     ob.atr=1;
7     return 0;
8 }
```

Exercice 9. *publique/privé PPI2.1*

Ce code compile-t-il ? si non pourquoi ?

```
1 class A{
2     int atr;
3 };
4 int main(void){
5     A ob;
6     return 0;
7 }
```

Exercice 10. *publique/privé PPI2.2*

Ce code compile-t-il ? si non pourquoi ?

```
1 class A{
2     A(void){}
3     int atr;
4 };
5 int main(void)
6 {
7     A ob; // piege
8     return 0;
9 }
```

3 Surcharge SI

Exercice 11. *surcharge SI3*

Ce code produit une erreur à la compilation, pourquoi ?

```
1 #include<iostream>
2 void f(char truc) {}
3 int* f(char a) {}
4 int main(void)
5 {
6     return 0;
7 }
```

Exercice 12. *surcharge SI1*

Qu'affiche ce code ?

```

1 #include<iostream>
2 void f(int a) { std::cout << "je_suis_f1_" << std::endl; }
3 int f(long a) { std::cout << "je_suis_f2_" << std::endl; }
4 void f(void) { std::cout << "je_suis_f3_" << std::endl; }
5 char f(char b, long d) { std::cout << "je_suis_f4_" << std::endl; }
6 int main(void)
7 {
8     f();
9     f(11);
10    f(11, 'a');
11    f(3);
12    return 0;
13 }

```

Exercice 13. *surcharge SI2*

Ce code produit une erreur à la compilation, pourquoi ?

```

1 #include<iostream>
2 void f(int a) { std::cout << "je_suis_f1_" << std::endl; }
3 int* f(char a) { std::cout << "je_suis_f2_" << std::endl; }
4 int main(void)
5 {
6     f(11);
7     return 0;
8 }

```

Exercice 14. *surcharge SI4*

Ce code compile correctement. Qu'affiche-t-il ?

```

1 #include<iostream>
2 struct A
3 {
4     A(){
5         std::cout<<"je_suis_A()"<<std::endl;
6     }
7     A(int b){
8         std::cout<<"je_suis_A(int)_avec_"<< b << std::endl;
9     }
10 };
11 int main(void)
12 {
13     A a;
14     A b{5};
15     A c={3};
16     A d(2);
17     A e(); // piege
18     return 0;
19 }

```

4 Constr/Destr CDI

Exercice 15. *construction/destruction CDI2*

Qu'affiche ce programme à l'exécution ?

```
1 #include<iostream>
2 class A{
3     int atr;
4     public:
5     A(void) { std::cout << "Construction" << std::endl; }
6     ~A(void) { std::cout << "Destruction" << std::endl; }
7 };
8 int main(void)
9 {
10     A ob;
11     return 0;
12 }
```

Exercice 16. *construction/destruction CDI4*

Qu'affiche ce programme à l'exécution ?

```
1 #include<iostream>
2 class A{
3     int atr;
4     public:
5     A(void) { std::cout << "Construction" << std::endl; }
6     ~A(void) { std::cout << "Destruction" << std::endl; }
7 };
8 int main(void)
9 {
10     A ob();
11     return 0;
12 }
```

Exercice 17. *construction/destruction CDI3*

Qu'affiche ce programme à l'exécution ?

```
1 #include<iostream>
2 class A{
3     int atr;
4     public:
5     A(int a)
6     {
7         atr=a;
8         std::cout << "C" << atr << "_";
9     }
10    ~A(void) { std::cout << "D" << atr << "_"; }
11 };
12 int main(void)
13 {
14     A ob0{0};
15     A ob1=1;
16     {
17         A ob2(2);
18         A ob3={3};
19     }
20     A ob4{4};
```

```

21     return 0;
22 }

```

Exercice 18. *construction/destruction CDI1*

Le code suivant produit une erreur. Pourquoi ?

```

1 #include<iostream>
2 struct A{
3     A(int a){std::cout << "Construction_A" << std::endl;}
4 };
5 int main(void){
6     A a;
7     return 0;
8 }

```

Exercice 19. *construction/destruction CDI5*

Qu'affiche ce code ? Que se passe si on decommente la ligne commentée ?

```

1 #include<iostream>
2 struct A{
3     A(void){
4         std::cout << "Construction_A1" << std::endl;}
5     A(long a){
6         std::cout << "Construction_A2" << std::endl;}
7     A(float a,int b=2){
8         std::cout << "Construction_A3" << std::endl;}
9 };
10 int main(void){
11     A ob0;
12     A ob1{};
13     A ob2{31};
14     A ob3(3.f);
15     A ob4{3,2};
16     //A ob5{3}; // (que se passe-t-il si on decommente la ligne ?)
17     return 0;
18 }

```

5 new, delete NDI

Exercice 20. *new/delete NDI1*

Qu'affiche l'exécution code suivant ?

```

1 #include<iostream>
2 struct A{
3     int a;
4     A(int p){
5         a=p;
6         std::cout << "C" << p << std::endl;
7     }
8     ~A(){
9         std::cout << "D" << a << std::endl;
10    }
11 };

```

```

12 int main(void){
13     A* p=new A(5);
14     A a(1);
15     delete p;
16 }

```

Exercice 21. *new/delete NDI6*

Qu'affiche l'exécution du code suivant ?

```

1 #include<iostream>
2 struct A{
3     int a;
4     A(int p){
5         a=p;
6         std::cout << "C" << p << std::endl;
7     }
8     ~A(){
9         std::cout << "D" << a << std::endl;
10    }
11 };
12 int main(void){
13     A* p=(A*) malloc (sizeof(A));
14     free(p);
15 }

```

Exercice 22. *new/delete NDI2*

Qu'affiche l'exécution code suivant ? Quel est le problème ?

```

1 #include<iostream>
2 struct A{
3     int a;
4     A(int p){
5         a=p;
6         std::cout << "C" << p << std::endl;
7     }
8     ~A(){
9         std::cout << "D" << a << std::endl;
10    }
11 };
12 int main(void){
13     A* p=new A(5);
14     A a(1);
15 }

```

Exercice 23. *new/delete NDI3*

Qu'affiche l'exécution code suivant ?

```

1 #include<iostream>
2 struct A{
3     int a;
4     A(int p){
5         a=p;
6         std::cout << "C" << p << std::endl;
7     }

```



```

8     ~A(){
9         std::cout << "D" << a << std::endl;
10    }
11 };
12 int main(void){
13     A a{2};
14     new(&a)A{4};
15 }

```

Exercice 24. *new/delete NDI4*

```

1 #include<iostream>
2 class A
3 {
4     int atr;
5     public:
6     A(void)
7     {
8         std::cout << "Bonjour" << std::endl;
9     }
10    ~A(void)
11    {
12        std::cout << "Au_revoir" << std::endl;
13    }
14 };
15 int main(void)
16 {
17     A ob;
18     A* obj2= new A();
19     A* obj3;
20     {
21         A obj4;
22         delete obj2;
23         obj2= new A();
24     }
25     A obj5;
26     return 0;
27 }

```

Exercice 25. *new/delete NDI5*

Ce code compile. En supposant que tout se passe bien, présente-t-il une fuite de mémoire à l'exécution ?

```

1 #include<iostream>
2 #include<cstdlib>
3 class A { };
4 int main(void)
5 {
6     int i;
7     A* ptr=(A*) malloc ( sizeof(A)*10 );
8     if (ptr==nullptr)
9     {
10         exit (1);
11     }

```

```

12     for ( i=0;i <5;i++)
13     {
14         new( ptr+i)A();
15     }
16     free( ptr );
17     return 0;
18 }

```

6 tableau TI

Exercice 26. *tableau TI1*

Ce code compile. Qu'affiche ce code ?

```

1 #include<iostream>
2 struct A{
3     int a;
4     A(){
5         std::cout << "C" << std::endl;
6     }
7     ~A(){
8         std::cout << "D" << std::endl;
9     }
10 };
11 int main(void)
12 {
13     A T[4];
14 }

```

Exercice 27. *tableau TI2*

Ce code ne compile pas. Pourquoi ?

```

1 #include<iostream>
2 struct A{
3     int a;
4     A(int p){
5         a=p;
6         std::cout << "C" << std::endl;
7     }
8     ~A(){
9         std::cout << "D" << std::endl;
10    }
11 };
12 int main(void)
13 {
14     A T[4]; // ici erreur
15 }

```

Exercice 28. *tableau TI3*

Ce code compile mais quel est son problème ?

```

1 #include<iostream>
2 struct A{
3     int a;

```

```
4     A(){
5         std::cout << "C" << std::endl;
6     }
7     ~A(){
8         std::cout << "D" << std::endl;
9     }
10 };
11 int main(void)
12 {
13     A* ptr=new A[10];
14     delete ptr;
15 }
```