



ECOLE NORMALE SUPÉRIEURE DE
L'ENSEIGNEMENT TECHNIQUE MOHAMMEDIA
DÉPARTEMENT MATHÉMATIQUES ET
INFORMATIQUE

24/04/2022

Compte rendu

Des travaux pratique de la
chapitre 6

Fonction amies

PRÉPARÉE POUR
TAFFAH ACHRAF

ENCADRÉ PAR
PR. KHALIFA MANSOURI



TABLE DES MATIERES

introduction	3
Partie pratique	3
1. Exemples de cour :	3
REMARQUE :	3
Exemple 1 :	4
Résultats :	4
Exemple 2 :	4
Résultats :	5
conclusion	5

INTRODUCTION

UNE FRIEND **FONCTION** EST UNE **FONCTION** QUI N'EST PAS MEMBRE D'UNE CLASSE, MAIS QUI A ACCES AUX MEMBRES PRIVES ET PROTEGES DE LA CLASSE. **LES FONCTIONS** FRIEND NE SONT PAS CONSIDEREES COMME DES MEMBRES DE CLASSE ; CE SONT DES **FONCTIONS** EXTERNES NORMALES POUR LESQUELLES DES PRIVILEGES D'ACCES SPECIAUX SONT ACCORDES.

PARTIE PRATIQUE

1. Exemples de cour :

REMARQUE :

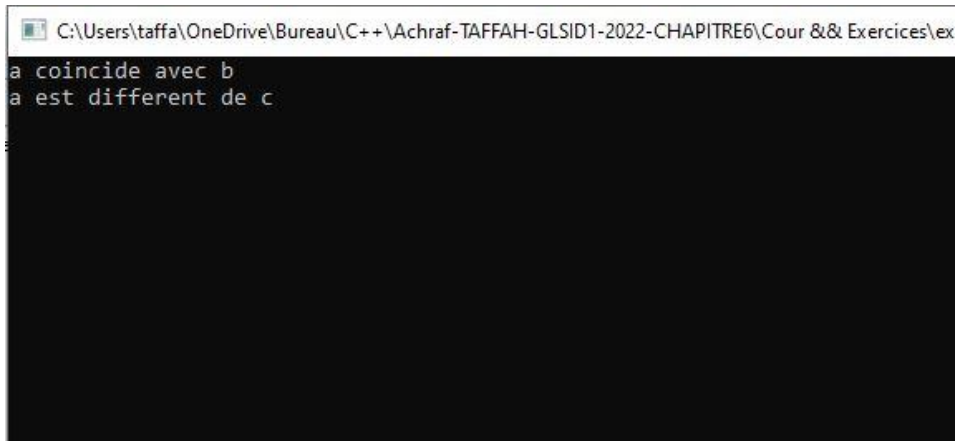
Une fonction amie est une fonction avec laquelle on pourra accéder aux membres privés d'une classe, d'une autre manière que l'accès à l'aide des fonctions membres.

Fonction indépendante amie d'une classe

EXEMPLE 1 :

```
2  #include<conio.h>
3  #include<string>
4  using namespace std;
5  class point{
6      int x,y;
7      public:
8          point(int abs=0,int ord=0){
9              x=abs;
10             y=ord;
11         }
12         friend int coincide(point,point);
13         //Déclaration de la fonction amie
14     };
15     int coincide(point p,point q){
16         if((p.x==q.x)&&(p.y==q.y))
17             return 1;
18         else
19             return 0;
20     }
21     int main(){
22         point a(4,0),b(4),c;
23         if(coincide(a,b))
24             cout<<"a coincide avec b \n";
25         else
26             cout<<"a est different de b\n";
27         if(coincide(a,c))
28             cout<<"a coincide avec c\n";
29         else
30             cout<<"a est different de c\n";
31         getch();
32         return 0;
33     }
```

RESULTATS :



```
C:\Users\taffa\OneDrive\Bureau\C++\Achraf-TAFFAH-GLSID1-2022-CHAPITRE6\Cour && Exercices\ex
a coincide avec b
a est different de c
```

Les autres situations d'amitié

EXEMPLE 2 :

```

3 #include<string>
4 using namespace std;
5 // Classe vecteur
6 // Surdefinition de l'operateur + par une fonction AMIE
7 class vecteur{
8     float x,y;
9     public:
10         vecteur(float,float);
11         void affiche();
12         friend vecteur operator+(vecteur,vecteur);
13 };
14 vecteur::vecteur(float abs=0,float ord=0){
15     x=abs;
16     y=ord;
17 }
18 void vecteur::affiche(){
19     cout<<"x = "<<x<<" y = "<<y<<"\n";
20 }
21
22 vecteur operator+(vecteur v,vecteur w){
23     vecteur res;
24     res.x=v.x+w.x;
25     res.y=v.y+w.y;
26     return res;
27 }
28 int main(){
29     vecteur a(2,6),b(4,8),c,d;
30     c=a+b; c.affiche();
31     d=a+b+c;d.affiche();
32     getch();
33     return 0;
34 }

```

RESULTATS :



```

C:\Users\taffa\OneDrive\Bureau\C++\Achraf-TAFFAH-GLSID1-2022-CHAPITRE6\Cour && Exercices\exemple2.exe
x = 6 y = 14
x = 12 y = 28

```

CONCLUSION

UNE FONCTION AMIE D'UNE CLASSE EST DONC DEFINIE EN DEHORS DE LA PORTEE DE CETTE CLASSE, MAIS ELLE A LE DROIT D'ACCEDER A TOUS LES MEMBRES PRIVES ET PROTEGES DE LA CLASSE. MEME SI LES PROTOTYPES DES FONCTIONS D'AMIS APPARAISSENT DANS LA DEFINITION DE LA CLASSE, LES AMIS NE SONT PAS DES FONCTIONS MEMBRES.