

The background of the slide is a spiral-bound notebook with a light beige, textured cover. The spiral binding is visible on the left side, with the wire looping through a series of holes. The text is centered on the page.

# Analyse et Programmation Orientées Objets / C++

## Classe Vecteur

# Header de la classe Vecteur (1)

---

```
# include <stdio.h>
# include <math.h>
# include <malloc.h>
const double EPSILON= 0.0001;
class Vecteur {
private :
double m_x;
double m_y;
public :
```

# Header de la classe Vecteur (2)

---

// Constructeurs et destructeur

//

Vecteur ();

Vecteur (double, double);

// Vecteur (const Vecteur&);

// ~Vecteur ();

# Header de la classe Vecteur (3)

---

```
// Accesseurs de consultation
//
inline double abscisse () {return m_x;}
inline double ordonnee () {return m_y;}
double norme ();
```

# Header de la classe Vecteur (4)

---

// Opérateurs

//

bool egal (const Vecteur&);

bool diff (const Vecteur&);

Vecteur\* add (const Vecteur&);

Vecteur\* sub (const Vecteur&);

Vecteur\* sub ();

Vecteur\* mul (double);

double prodS (const Vecteur&);

# Header de la classe Vecteur (5)

---

```
// Services
```

```
//
```

```
bool colineaire (const Vecteur&);
```

```
bool orthogonal (const Vecteur&);
```

```
bool memeSens (const Vecteur&);
```

```
char* toString  ();
```

```
};
```

# Constructeurs

---

```
# include "Vecteur.h "
```

```
Vecteur::Vecteur () {  
    m_x= 0;  
    m_y= 0;  
}
```

```
Vecteur::Vecteur (double x, double y) {  
    m_x= x;  
    m_y= y;  
}
```

# Constructeur de copie et destructeur

---

**Le constructeur de copie est généré automatiquement !**

**Le destructeur est généré automatiquement !**

**Génération non valide dans les cas d'au moins un attribut alloué dynamiquement**



# Exemple d'opérateurs (1)

---

```
# include "Vecteur.h "
```

```
Vecteur* add (const Vecteur& V) {
```

```
double x, y;
```

```
    x= abscisse () + V.abscisse ();
```

```
    y= ordonnee () + V.ordonnee ();
```

```
    return new Vecteur (x, y);
```

```
}
```

# Exemple d'opérateurs (2)

---

```
# include "Vecteur.h "
```

```
Vecteur* mul (double k) {  
    double x, y;
```

```
        x= k*abscisse ();
```

```
        y= k*ordonnee ();
```

```
        return new Vecteur (x, y);
```

```
}
```

# Exemple de services

---

```
# include "Vecteur.h "
```

```
bool colineaire (const Vecteur& V) {  
    double d1= abscisse()*V.ordonnee();  
    double d2= ordonnee()*V.abscisse();  
  
    return fabs(d1-d2) < EPSILON;  
}
```

**Problème permanent des erreurs d'arrondis !**