

Programmation Orientée Objet (C++) :

Variables et méthodes de classes

Surcharge d'opérateurs

Jamila Sam

Laboratoire d'Intelligence Artificielle
Faculté I&C



Surcharge d'opérateurs



```
class Classe {
    ...
    type_retour operatorOp(type_argument); // prototype de l'opérateur Op
    ...
};
```

```
// définition de l'opérateur Op
type_retour Classe::operatorOp(type_argument) { ... }
```

```
// opérateur externe
type_retour operatorOp(type_argument, Classe&) { ... }
```

Quelques exemple de prototypes :

```
bool operator==(Classe const&) const; // ex: p == q
bool operator<(Classe const&) const; // ex: p < q
Classe& operator=(Classe const&); // ex: p = q
Classe& operator+=(Classe const&); // ex: p += q
Classe& operator++(); // ex: ++p
Classe& operator*=(const autre_type); // ex: p *= x;
Classe operator-(Classe const&) const; // ex: r = p - q
Classe operator-() const; // ex: q = -p;
```

```
// opérateurs externes
ostream& operator<<(ostream&, Classe const&);
Classe operator*(double, Classe const&);
```

Vidéos, transparents et quiz

<https://www.coursera.org/learn/programmation-orientee-objet-cpp/home/week/3>

📅 Semaine 3

Pour préparer le prochain cours

► Vidéos et quiz du MOOC semaine 4 :

- Héritage : concepts [15 :20]
- Héritage : droit protégé [10 :51]
- Héritage : masquage [8 :25]
- Héritage : constructeurs (2) [11 :49]
- Héritage : constructeurs (2) [11 :39]
- Copie profonde [16 :33]

► Le prochain cours :

- de 14h15 à 15h (résumé et quelques approfondissements)