



# Surcharge d'opérateurs



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

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

// 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, Classe const&); // ex: r = p - q
Classe operator-() const; // ex: q = -p;

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