Surchage des opérateurs

Question 1 :

L'opérateur _____ est utilisé par C++ pour attribuer un objet à un autre objet.

=

Question 2 :

Offrir la possibilité à un opérateur intégré C++ de travailler avec des types définis par l'utilisateur s'appelle _____.

surcharge de l'opérateur

Question 3 :

La surcharge de l'opérateur d'affectation C++ doit être effectuée lorsque la classe contient _____.

pointeurs bruts

Question 4 :

Si aucun opérateur d'affectation surchargé n'est fourni par le programmeur, C++ fournira une affectation par défaut qui fait _____.

affectation des membres

Question 5 :

Pour surcharger l'opérateur d'insertion afin que vous puissiez insérer vos objets de classe Test dans un flux de sortie, vous implémenterez une fonction avec le prototype suivant :

```
std::ostream &operator<<(std::ostream &os, const Test &obj);
```

Question 6 :

Lors de la surcharge d'un opérateur C++, laquelle des affirmations suivantes est vraie ?

L'arité de l'opérateur ne peut pas changer. Les opérateurs non C++ ne peuvent pas être surchargés. Certains opérateurs C++ ne peuvent pas être surchargés. L'opérateur d'affectation doit être surchargé en tant que fonction membre

Question 7 :

La plupart des opérateurs C++ peuvent être surchargés en tant que _____ ou _____.

fonctions membres, fonctions non membres

Question 8 :

Si nous surchargeons les opérateurs relationnels C++ tels que `==, !=, <, >, <=, >=` nous devrions renvoyer le type _____ de la fonction.

bool

Question 9 :

Souvent, les opérateurs de surcharge implémentés en tant que fonctions non membres sont déclarés comme fonctions _____.

friend

Question 10 :

Quel est le prototype correct pour l'opérateur Move Assignment pour une classe nommée Test ?

```
Test &Test::operator=(Test &&rhs);
```