**Les fonctions lambda en C++. Capture et fermeture**: Les fonctions lambda de C++ (ajoutées dans la version 2011 du standard) fournissent un mécanisme efficace pour définir localement une fonction qui peut ensuite être utilisée, par exemple, comme prédicat dans un algorithme. Au moment de construire une fonction lambda, le langage fournit différents mécanismes pour rendre accessibles des valeurs de variables et de paramètres qui seront utilisés lors de l'exécution de la fonction lambda. On appelle ce mécanisme la "capture" des variables et des paramètres.

Les variables peuvent être capturées par référence ou par valeur et sont ajoutées à ce qu'on appelle la "classe de fermeture" associée à la fonction lambda. Décrivez l'utilité de la classe de fermeture d'une fonction lambda, expliquez la différence entre les différents modes de capture et précisez le ou les modes qui devraient être privilégiés, en décrivant les problèmes potentiels qui peuvent se produire lorsque le mode de capture est mal choisi. Selon vous, y-a-t-il des règles qui devraient être suivies de façon systématique lorsque des variables ou des paramètres sont capturés dans une fonction lambda ? Si oui lesquelles ?