Techniques de l’informatique – programmation de jeux vidéo

420-V32-SF Programmation de jeux vidéo III

Introduction au C++

Partie 1 : Généralités sur le C++

Référez-vous aux diapositives « Du C# au C++ » pour répondre aux questions suivantes.

1. L’instruction suivante

**while(1)**

{

//… instructions valides

}

est valide en C++ (mais non en C#!). Expliquez en vos mots pourquoi.

Parce qu’en C++ il est possible d’effectuer une conversion de booléen (tout nombre autre que 0 donnera TRUE tandis que les autres donneront FALSE). Cette opération dans un WHILE n’est cependant pas recommandée, car convertir un nombre en booléen implique une opération de conversion à chaque rappel de la boucle.

1. A quoi sert l’instruction setLocale?

Cette méthode sert à user du jeu de caractères par défaut du clavier. Utile pour s’ajuster à la langue.

1. A quoi sert l’instruction system("Pause");?

En mode DEBUG, il demeure utile de pouvoir éviter la fermeture de la console. Cette méthode est donc utile pour arrêter l’exécution de la console au moment voulu. Cependant, cette instruction n’est pas toujours fonctionnelle sur les autres plateformes.

1. Le mot clé static existe en C++ et a *à peu près* la même utilité qu’en C#. Par contre, on peut l’utiliser d’une nouvelle manière. Expliquez à quoi il peut servir à l’intérieur d’une fonction ou d’une méthode.

À l’intérieur d’une fonction ou d’une méthode, le mot clé statique donnera une valeur à sa variable qu’à la première instanciation. Ceci peut être utile pour connaître la première valeur envoyée lors du premier appel de la fonction par exemple.

1. Le passage de paramètres en C++ ressemble *un peu* à celui du langage C#. Les types de base sont passés par valeur. Dites si les types complexes (ex. classes) sont également passés par valeur. **Soyez précis dans votre réponse**.

Effectivement, en C++, les types de base comme les type complexes sont passés par valeur, ce qui signifie que le code copiera aussi les classes.

1. A quoi sert l’utilisation du & lors du passage d’un paramètre à une fonction ou à une méthode?

L’utilisation du & lors du passage d’un paramètre à une fonction ou à une méthode envois la référence à la ou fonction ou à la méthode en question.

1. À quoi sert l’instruction suivante :

#define NB\_PERSONNAGES\_DANS\_JEU 100

Cette instruction, bien que non recommandée remplace la valeur dans toutes les instances portant le nom « NB\_PERSONNAGES\_DANS\_JEU pour 100. Cet action imposera une valeur de précompilation, qui se rapproche le plus dans le C++ des constantes en C#.

1. (Plus difficile) Expliquez le « fonctionnement » de cette instruction lors de la compilation.

Juste avant la compilation, comme expliqué précédemment, toutes les valeurs portant le même noms que celle entrée seront remplacées par la valeur entrée.

Partie 2 : Conversion de projets C# vers C++

Récupérez l’archive ExercicesConversionProgramme sur LÉA puis répondez aux questions suivantes.

1. Créez-vous une nouvelle solution puis procédez à la « traduction » en C++ des projets se trouvant dans la solution ExercicesConversionProgramme.

**Notes :**

* 1. Les nombres aléatoires sont générés en C++ à l’aide des fonctions srand et rand déclarées dans le fichier stdlib.h. Recherchez sur Google de quelle manière elles doivent être utilisées.
  2. Les fonctions mathématiques sont déclarées dans le fichier cmath.

Partie 3 : Questions diverses sur les projets convertis en C++ de la partie 2

1. Dans le fichier main.cpp du projet FonctionsDiverses, vous avez dû instancier des points. Dites ce qui distingue les deux instructions suivantes :

Point p1;

et

Point p1();

**Note** : je n’ai jamais écrit qu’elles étaient toutes deux valides!!!

1. Supposons que vous aviez voulu écrire une fonction visant à déterminer si un point reçu en paramètre est dans un quadrant également reçu en paramètre. La déclaration de la fonction en question aurait pu être :

bool EstDansQuadrant(Point p, int quadrant);

Donnez un désavantage à cette déclaration. Orientez-votre réponse en fonction du mode de passage de paramètre utilisé.

1. La déclaration de la méthode InitialiserTableau du projet TableauxStaticConstantes aurait pu (ou dû!) être la suivante :

void InitaliserTableau(int tab[]);

Dites quel est le mode de passage de paramètre utilisé pour le paramètre tab.

1. Quel est l’équivalent de la propriété .Length pour un tableau en C++?