## Contrôle de C++

Nom:	Prén	om:	
1. Citez et décrivez 4 d'un logiciel.	qualités fondamentales r	recherchées dans la co	enception et le développement
2. Dans la déclaration de const à la fin de son pa		sse C++, à quoi corre	espond l'utilisation du mot-clé

On souhaite obtenir la représentation binaire en complément à 2 de nombres entiers décimaux de type long long int sur un nombre variable de bits. Par exemple, sur 8 bits, la représentation binaire de l'entier décimal 20 est 00010100, et celle de -114 est 10001110. De même, sur 16 bits, on a 1024 = 000001000000000000 et -1 = 111111111111111111. Le nombre de bits pour la représentation d'un nombre décimal pourra donc varier de 1 à sizeof (long long int)\*8. D'autre part, l'entier à convertir sur b bits doit être compris entre  $-2^{b-1}$  et  $2^{b-1}-1$ . Pour les vérifications de validité, vous pourrez utiliser assert.

3. Dans le fichier décimal.hpp, écrivez la classe décimal avec
— la constante membre <i>privée</i> maxBits avec sa valeur;
<ul> <li>— la variable membre privée n de type long long int;</li> <li>— un constructeur public qui initialise la variable membre n;</li> </ul>
— l'en-tête de la méthode <i>publique</i> getN qui renvoie la valeur décimale de n;
— l'en-tête de la méthode <i>publique</i> versBinaire qui renvoie sous forme d'une chaîne de caractère (std::string) la représentation binaire de la variable membre n. Le nombre de bits est passé
en paramètre.
en parametre.

▶ 4. Dans le fichier décimal.cpp, programmez la méthode versBinaire. Indication : il s'agit de faire la décomposition de this->n en base 2. Notez que si this->n est négatif, il sera judicieux de décomposer son complément sur la plus grande valeur décimale sur le nombre de bits demandés. Par exemple, le complément de −114 sur 8 bits est −114 + 256 = 142. La décomposition binaire de 142 est 10001110, qui est donc la représentation binaire de −114 sur 8 bits. Pour créer la chaîne de caractères résultat renvoyée par la méthode versBinaire, utilisez l'opérateur + et la méthode insert de la classe std::string (voir annexe). Enfin, vous ne devez pas utiliser la fonction pow (ou toute autre fonction) pour calculer des puissances de 2.


## ▶ 5. Écrivez la fonction main qui :

- lit sur l'E/S un nombre de bits  $\in [2, \text{ sizeof(long long int)*8}].$
- déclare la variable table, un vecteur de <u>pointeurs</u> sur décimal. La taille du vecteur est égale au nombre maximum de décimaux sur le nombre de bits lu;
- initialise le vecteur avec les décimaux compris entre  $-2^{b-1}$  et  $2^{b-1}-1$ , où b est le nombre de bits lu sur l'E/S
- parcourt toute la table pour écrire sur la S/S tous les décimaux avec, pour chacun, sa représentation binaire.

Par exemple, si on lit sur l'E/S l'entier 8, l'exécution de la fonction main écrira 256 nombres surt la S/S:

nb bits = 8 -128 = 10000000 -127 = 10000001 -126 = 100000102 = 11111111 0 = 00000000 1 = 00000000 1 125 = 01111101 126 = 01111110 127 = 01111111			