*Remarque : les différentes conversions doivent utiliser les calculs et non des fonctions toutes faites comme « bitset »*

1. Créer deux fonctions de conversion en C++ : binaire/décimal, décimal/binaire. Afficher un menu permettant de choisir parmi ces deux opérations.

Exemples de calculs décimal vers binaire :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Diviseur | Dividende |  |  |  |
|  | 5 | 2 |  |  |  |
| reste1 --> | **1** | 2 | 2 |  |  |
|  | reste2 --> | **0** | 1 | 2 |  |
|  |  | reste3 --> | **1** | 0 |  |
|  |  |  |  |  |  |
| résultat = | reste3 | reste2 | reste1 | =101 |  |
|  |  |  |  |  |  |
|  | 9 | 2 |  |  |  |
| reste1 --> | **1** | 4 | 2 |  |  |
|  | reste2 --> | **0** | 2 | 2 |  |
|  |  | reste3 --> | **0** | 1 | 2 |
|  |  |  | reste4 --> | **1** | 0 |
|  |  |  |  |  |  |
| résultat = | reste4 | reste3 | reste2 | reste1 | = 1001 |
|  |  |  |  |  |  |
|  | 11 | 2 |  |  |  |
| reste1 --> | 1 | 5 | 2 |  |  |
|  | reste2 --> | 1 | 2 | 2 |  |
|  |  | reste3 --> | 0 | 1 | 2 |
|  |  |  | reste4 --> | 1 | 0 |
|  |  |  |  |  |  |
| résultat : | reste4 | reste3 | reste2 | reste1 | = 1101 |

**Pour la conversion décimale vers binaire:**

On commence par entrer le nombre décimal à convertir au clavier.

Calcul 1 :

Pour calculer le reste entier on utilise l’opérateur modulo : **%**

Exemple : 5%2 (reste de 5/2).

Calcul 2 :

Pour calculer la partie entière d’une division on utilise la fonction **floor** :

Exemple : cout<< floor(5/2) ;

Tester cette fonction avec au moins quatre calculs pour bien comprendre son fonctionnement.

Pour pouvoir utiliser la fonction « floor », on doit inclure la bibliothèque cmath de la façon suivante : #include <cmath>

Rappel : ces deux calculs doivent être fait tant que le quotient est différent de zéro

Pour vous aider, faites d’abord sur papier une conversion.

**Pour la conversion binaire vers décimale :**

Le nombre entré au clavier sera une chaîne de caractères.

Remarque : une chaîne de caractère est aussi un tableau de caractères ; on peut accéder à un caractère comme pour un tableau :

string chaine = ‘’salut’’ ;

cout << chaine[0] ; 🡪 affichera ‘s’

cout << chaine[1] ; 🡪 affichera ‘a’

string chaine1 = ‘’1011’’ ;

cout << chaine1[0] ; 🡪 affichera ‘1’

cout << chaine1[1] ; 🡪 affichera ‘0’

Il faudra convertir chaque élément de la chaîne en un nombre. L’opération à effectuer sera :

int nombre ;

string chaine2 = ‘’1234’’ ;

nombre = chaine2[0] – ‘0’ ; 🡪 stocke le chiffre 1 dans nombre

nombre = chaine2[1] – ‘0’ ; 🡪 stocke le chiffre 2 dans nombre

nombre = chaine2[2] – ‘0’ ; 🡪 stocke le chiffre 3 dans nombre

La fonction puissance : pow(3,2) calcule 3 puissance deux soit trois au carré soit 3².

1. Faire de même avec les conversions décimal vers hexadécimal et inversement.