

KOUROUMA
Idriss

NFO4
Sujet B5 - Somme de grands entiers

Spécifications:

Dans le cadre de l'UE NFO4 nous devons réaliser un projet. Ce dernier porte sur la création d'un algorithme permettant d'effectuer la somme ^{ou la soustraction} de très grands entiers (au plus 80 chiffres).

Données : Les deux entiers avec au plus 80 chiffres, leur signe ('+' ou '-') et l'opération voulue ('+' ou '-'), saisies au clavier.

Résultats : La somme ou soustraction (selon la demande) des deux entiers. Le résultat du calcul affiché à l'écran.

Analyse:

Les deux entiers vont être contenus chacun dans un tableau de 80 caractères : TabNb1 et TabNb2.
Leur saisie s'effectuera de la même manière : valeur absolue saisie en D.
On affecte à la première case du tableau (a:TabNb1[0]) la valeur du premier caractère saisi par l'utilisateur. Puis on initialise à 0 l'indice correspondant à l'entier en cours de saisie (i:Nb1 ou Nb2) et on lance une boucle indéterminée (tant qu) qui s'arrête lorsque le caractère lu (TabNb1[i:Nb1]) est égal à '\0'.
On incrémente l'indice puis on lit.
Les deux tableaux sont alors remplis. On demande le signe des 2 entiers et on les stocke dans les variables Sign1 et Sign2.

Ensuite, nous allons initialiser 2 variables qui serviront les

des valeurs :

- fin qui représente la longueur du plus grand entier.
- PlusGrand qui désigne l'entier le plus grand (1 ou 2).

Pour cela, on va utiliser iNb1 et iNb2 qui représente le nombre de caractères dans chaque tableau.

Si il y a plus de caractère dans le premier tableau que dans le deuxième ($iNb1 > iNb2$), fin prend la valeur iNb1 et PlusGrand prend la valeur 1.

Sinon fin prend la valeur 2 et si il y a plus de caractère dans le tableau 2, PlusGrand prend la valeur 2, sinon (le nombre de caractères est identique) on met une boucle indéterminée qui s'arrête lorsque les valeurs sont différentes ou si l'indice i est supérieur à fin ($i > fin \rightarrow$ les entiers sont les mêmes). Cette boucle permet de comparer un à un les caractères de chaque tableau.

On vérifie pourquoi on est sorti de la boucle : Si le dernier caractère comparé est plus grand dans le premier tableau alors PlusGrand = 1 sinon PlusGrand = 2 (même si ils sont égaux).

Ensuite, on demande à l'utilisateur quelle opération il veut faire, on stock la réponse dans la variable calcul.

Avant de rentrer dans les conditions opératoires, on initialise les variables suivantes :

retourne : qui servira en cas de retour dans le calcul. Initialisé à 0.

i : initialisé à fin, qui permet de se déplacer entre la fin du premier tableau et 0.

ic1 et ic2 : initialisés respectivement à $iNb1 - 1$ et $iNb2 - 1$ (car les tableaux commencent à 0), serviront à se déplacer dans TabNb1 et TabNb2.

Les additions et soustractions s'effectuent de manière différente selon l'opération demandée et le signe des deux entiers.

Toutes les possibilités sont réunies dans le tableau suivant :

Opération demandée par l'utilisateur	Signe 1	Signe 2	Opération effectuée	Signes
Addition	+	+	Addition	Positif
Soustraction	+	-	Addition	Positif
Addition	-	-	Addition	Négatif
Soustraction	-	+	Addition	Négatif
Addition	+	-	Soustraction: $1-2$ $2-1$	Positif Négatif
Soustraction	+	+	Soustraction: $1-2$ $2-1$	Positif Négatif
Addition	-	+	Soustraction: $1-2$ $2-1$	Négatif Positif
Soustraction	-	-	Soustraction: $1-2$ $2-1$	Négatif Positif

Légende : - Opération effectuée = opération que le programme va dire de faire

- rouge pour $Nb1 \geq Nb2$

- noir pour $Nb2 \geq Nb1$

Pour les mêmes opérations effectuées (addition)

Pour les mêmes opérations effectuées (soustraction)

Il y a donc 8 conditions, 4 amenant à une addition et 4 amenant à une soustraction.

Pour l'addition : on vérifie si l'une des 4 conditions sont remplies à l'aide d'un Si. On initialise ires (qui permet de se déplacer dans le tableau des résultats, TabRes) avec la valeur fin. On effectue l'addition avec une boucle déterminée avec l'indice i qui va de fin à 1 par pas de -1.

Si l'un des deux tableaux n'a plus de chiffre, le tableau des résultats prend les valeurs des chiffres restants de l'autre tableau (Boucle Pour si $ic1 < 0$, Boucle Pour si $ic2 < 0$)

Puis on passe à la case suivante. A la fin, si il reste une retenue (retenue = 1) alors la case 0 du tableau des résultats prend la valeur 1 et on affiche le tableau avec une boucle Pour de 0 à fin. Sinon (retenue = 0), on

affiche avec une boucle Pour de 1 à fin.

exemple :

1 2 3 4

fin = 4

+ 9 3 8 1

2 1 5 5

retourne = 0

TabRes :

[0] [1] [2] [3] [4]

↑ ↑ ↑ ↑ ↑
1 1 1 5 5

retourne = 1

Si le calcul remplit les conditions Pour lequel SignRes est négatif (voir tableau) alors on affiche que le résultat est négatif.

Sinon on affiche juste le résultat.

Pour la soustraction : on vérifie avec un si que le calcul remplit une des 4 conditions. Et selon si le premier entier est le plus grand ou non (Plus Grand = 1) alors on soustrait 1 par deux si c'est le cas sinon 2 par 1 avec une boucle Pour de fin-1 à 0 (pas besoin de laisser de place pour une retenue). Selon le calcul on affiche que le résultat est négatif ou pas.