

SECTION : SCIENCES DE L'INFORMATIQUE

EPREUVE : ALGORITHMIQUE ET PROGRAMMATION

DURÉE : 1 H

NOMBRE DE PAGES : 1

COEF : 3

Important :

- 1) Une solution modulaire au problème posé est exigée.
- 2) Dans le dossier C:\Bac2010\4SI\ALGO, enregistrer tous les fichiers solutions au problème posé. Attention, tout fichier que vous enregistrerez en dehors de votre dossier de travail ne sera ni recherché, ni pris en compte et sera systématiquement détruit.

Problème :

Le type entier long (LongInt) du langage Pascal ne permet pas par exemple de faire une addition de deux entiers dont le résultat dépasse la valeur 2 147 483 647.

Pour dépasser cette limite, on se propose d'écrire un programme nommé " *Grande_Addition* ", qui fait une addition exacte de deux entiers naturels dont le résultat peut avoir jusqu'à 255 chiffres. Pour cela, la saisie des entiers se fera sous forme de chaînes de caractères. Ces chaînes sont formées seulement par des chiffres décimaux ("0" .. "9").

Pour faciliter l'opération, le programme ajustera ces deux chaînes de telle façon qu'elles auront la même longueur. Cette opération consiste à ajouter des 0 à gauche de la plus courte des chaînes pour qu'elle soit de même longueur que l'autre.

On passe ensuite à l'opération d'addition et le résultat obtenu est mis dans une troisième chaîne qui sera affichée.

Algorithme du programme principal :

- 0) Début Grande_Addition
- 1) Proc Saisir(A)
- 2) Proc Saisir(B)
- 3) Proc Completer_Par_Des_Zeros(A,B)
- 4) C ← Fn Additionner (A,B)
- 5) Ecrire ("La somme de ",A, " et ", B, " est : " , C)
- 6) Fin Grande_Addition

Question

1. Dans un fichier **VotreNum_AD1.pas**, traduire l'algorithme ci-dessus, déclarer les variables globales nécessaires et les entêtes des modules (sans préciser leurs traitements).
2. Dans un fichier **VotreNum_AD2.pas**, compléter la solution, en définissant les modules appelés.

Bon travail !

Barème :

- Q1 Traduction : 4 points
Q2 : Proc saisir : lecture 2 points + contrôle 3 points
Proc Completer : 3 points
Fn Additionner : 8 points