

## Exercice 6, révision chapitres 1 à 6

### OBJECTIF

A l'issue de la réalisation de cet exercice, les étudiants doivent être capable de traduire la donnée en un ensemble de structogramme de Jackson puis de réaliser pratiquement le problème avec une application console avec le Visual studio 2010.

Réalisation du programme sur la base d'un fichier canevas.

### CONSIGNE DE TRAVAIL

Le travail se découpe en trois étapes :

#### 1 ETUDE ET REALISATION DES STRUCTOGRAMMES

Comprendre et traduire en un structogramme de Jackson le traitement à effectuer pour le testA et le testB.

#### 2 PREPARATION DU PROJET

L'exercice 2 sert de mode d'emploi pour la création du projet.

- Créez un répertoire sur le disque local par exemple C:\LOGA\_C\Nom (Nom étant votre nom de famille) ou dans les documents locaux.
- Lancez le "Microsoft Visual Studio 2015"
- Créez un nouveau projet :
  - Projets Visual C++
  - Application console Win32 (Dans la fenêtre modèle)
  - Sous emplacement: introduisez C:\LOGA\_C\Nom
  - Décocher "Créer le répertoire pour la solution"
  - Sous Nom : **Ex6**
- Paramètre de l'application :
  - Application Console
  - Projet Vide
- **Copie du fichier Ex6.c :**

Il faut copier le fichier Ex6.C de K:\ES\Maitres-Eleves\SLO\Modules\SL121 LOGA\Exercices\Ex6 dans le répertoire Ex6. (Usage de l'explorateur Windows)

- Ajout du fichier Ex6.c :

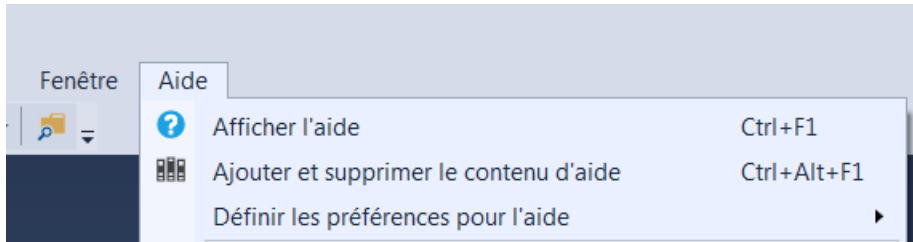
Depuis le Visual Studio 2015, sélectionnez "**Fichiers Sources**", avec un clique-droit obtenez le menu pour Ajouter un élément existant.
- Test de compilation : vous devez obtenir :

Génération : 1 a réussi, 0 a échoué, 0 a été ignore

Attention

- Test travail sans réseau : ☹️ veuillez débrancher le câble réseau !

👉 Il faut essayer d'afficher l'aide.



Si l'aide ne s'affiche pas, suivre les indication orale pour la configurer. Remarque risque de ne pas avoir les droits pour configurer.

### 3 REALISATION DU PROGRAMME

Le programme traite 2 cas séparés. Le système de sélection des tests est fourni ainsi que la saisie de ValA et ValB, il faut ajouter les déclarations des variables supplémentaires dont vous avez besoins.

Il faut réaliser les actions correspondant au 2 cas décrit ci-dessous en se basant sur les structogrammes établis.

Il faut encore afficher (en adaptant), une seule fois au lancement du programme :

#### **Exercice 6 : Nom Prenom**

##### **ACTION TEST A**

Vous obtenez dans la variable ValA la valeur introduite par l'utilisateur.

Si ValA est plus grande que 9 il faut la limiter à 9 et afficher :

"TestA ValA limitée à 9 !"

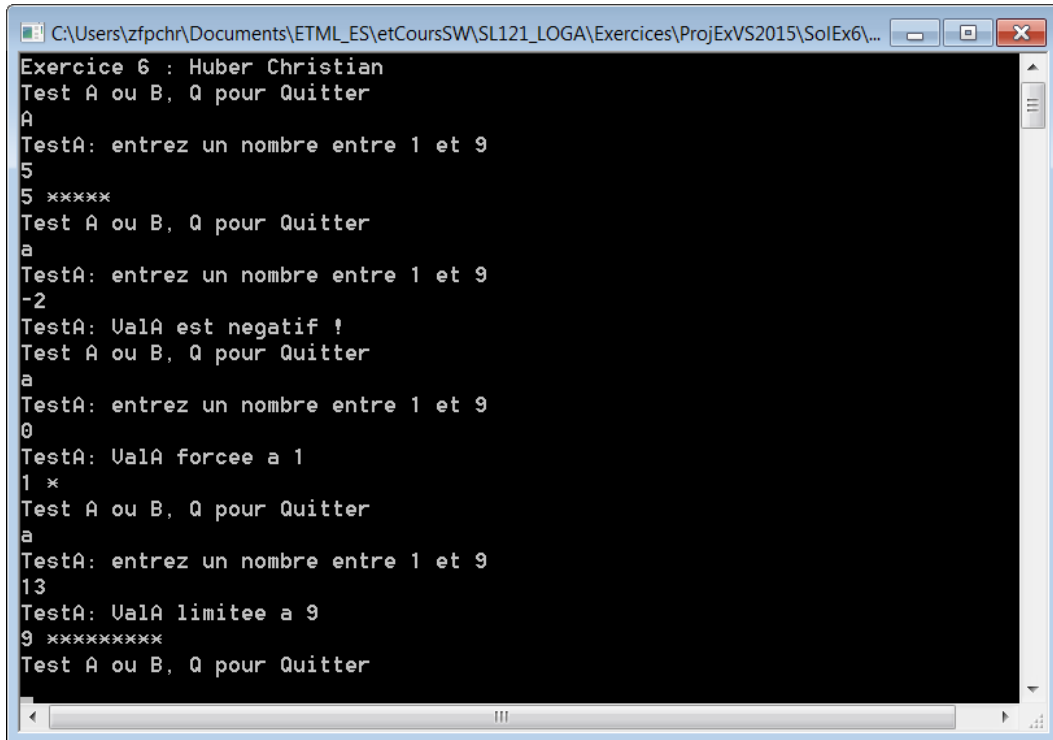
Si ValA vaut 0 il faut imposer 1 et afficher : "TestA: ValA forcée à 1 !"

Si ValA est positif, il faut exécuter l'affichage de la valeur de ValA suivit du nombre d'\* correspondant. Exemple pour ValA = 4

**4 \*\*\*\***

Si ValA est négatif, il faut afficher : "TestA: ValA est négatif !" et ne rien exécuter.

Exemple de résultat à obtenir :



```
C:\Users\zfpchr\Documents\ETML_ES\etCoursSW\SL121_LOGA\Exercices\ProjExVS2015\SolEx6\...
Exercice 6 : Huber Christian
Test A ou B, Q pour Quitter
A
TestA: entrez un nombre entre 1 et 9
5
5 *****
Test A ou B, Q pour Quitter
a
TestA: entrez un nombre entre 1 et 9
-2
TestA: ValA est negatif !
Test A ou B, Q pour Quitter
a
TestA: entrez un nombre entre 1 et 9
0
TestA: ValA forcee a 1
1 *
Test A ou B, Q pour Quitter
a
TestA: entrez un nombre entre 1 et 9
13
TestA: ValA limitee a 9
9 *****
Test A ou B, Q pour Quitter
```

Aide printf : si on n'introduit pas de `\n`, l'affichage continue sur la même ligne.  
Pour effectuer uniquement un retour de ligne : `printf("\n");`

## ACTION TEST B

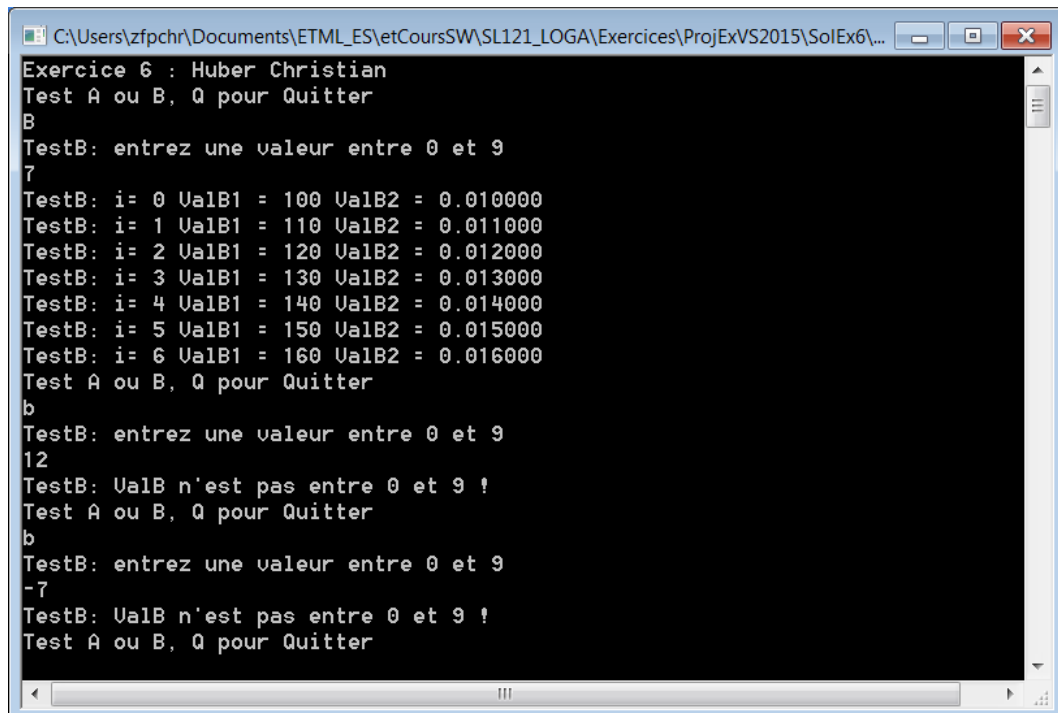
Vous devez déclarer une variable ValB1 de type short et une variable ValB2 de type double.

La valeur ValB fournie par l'utilisateur (entre 0 et 9) doit être contrôlée. Si ValB n'est pas comprise entre 0 et 9 il faut afficher : "TestB : ValB n'est pas entre 0 et 9 !" et ne pas réaliser d'action. Si ValB est correcte il faut réaliser une boucle variant de 0 à ValB-1, il faut afficher sur chaque ligne :

"TestB: i= v ValB1 = vvvv ValB2 = v.vvvvvvvv".

Avec ValB1 = 100 + 10 \*i et ValB2 = ValB1 / 10000.

Exemple de résultat à obtenir :



```
C:\Users\zfpchr\Documents\ETML_ES\etCoursSW\SL121_LOGA\Exercices\ProjExVS2015\SolEx6\...
Exercice 6 : Huber Christian
Test A ou B, Q pour Quitter
B
TestB: entrez une valeur entre 0 et 9
7
TestB: i= 0 ValB1 = 100 ValB2 = 0.010000
TestB: i= 1 ValB1 = 110 ValB2 = 0.011000
TestB: i= 2 ValB1 = 120 ValB2 = 0.012000
TestB: i= 3 ValB1 = 130 ValB2 = 0.013000
TestB: i= 4 ValB1 = 140 ValB2 = 0.014000
TestB: i= 5 ValB1 = 150 ValB2 = 0.015000
TestB: i= 6 ValB1 = 160 ValB2 = 0.016000
Test A ou B, Q pour Quitter
b
TestB: entrez une valeur entre 0 et 9
12
TestB: ValB n'est pas entre 0 et 9 !
Test A ou B, Q pour Quitter
b
TestB: entrez une valeur entre 0 et 9
-7
TestB: ValB n'est pas entre 0 et 9 !
Test A ou B, Q pour Quitter
```

## 4 SAUVEGARDE FINALE

Lorsque le programme est au point, vous pouvez rebrancher le câble réseau et copier l'entier de votre projet à l'emplacement habituel sur votre disque réseau H:

Remarque : cette manœuvre sert d'entraînement pour l'épreuve pratique.