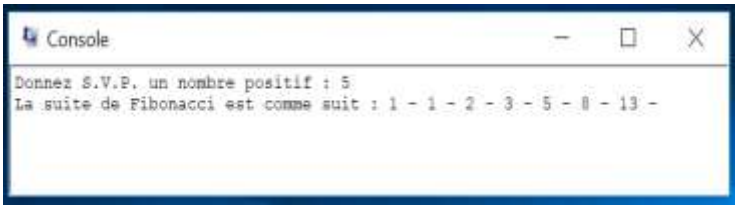
Université des sciences et de la technologie d'Oran – MB
Faculté des Mathématiques et de l'Informatique - Département d'informatique
2ème Année LMD - S3 – 2022/2023

TP N°3 – Architecture des ordinateurs
Les procédures / fonctions

Exercice 1 :

Revenir sur l'exercice 6 de la fiche de TP2 permettant d'écrire le code MIPS pour calculer les N premiers éléments d'une suite de Fibonacci comme indiqué dans la fenêtre ci-dessous.

Exemple d'exécution



1. Modifier le programme déjà fait, en remplaçant la partie du code permettant d'afficher un séparateur entre chaque paire de termes par une procédure « **AfficheSeparateur** » permettant d'afficher le séparateur « **-** »
2. Mettre à jour le programme en remplaçant la procédure « **AfficheSeparateur** » par une procédure « **AfficheChaine** » permettant d'afficher une chaine de caractères qui doit être passée en paramètre avant d'appeler cette procédure.
3. Ajouter, à la dernière version du programme, une procédure « **LireEntier** » permettant de lire un entier et faire appel à cette procédure aux endroits adéquats.
4. Ajouter, à la dernière version du programme, une procédure « **AfficheEntier** » permettant d'afficher un entier et faire appel à cette procédure aux endroits adéquats.

Exercice 2 :

Soit le programme « **C** » suivant permettant de faire la somme de deux entiers en appelant une fonction « **addition** » à deux paramètres (**a** et **b**).

```
1   int addition (int a , int b) ;
2   int main()
3   {
4   int x = addition(5 , 15) ;
5   return 0 ;
6   }
7   int addition (int a , int b)
8   {
9   return a + b ;
10  }
```

Ecrire le code MIPS correspondant en respectant les conventions de passage de paramètres **a** et **b** tels que **a=5** et **b=15** lors de l'appel de cette fonction.

Exercice 3 :

Écrire en assembleur MIPS le code permettant de :

a. Définir la fonction **int impair(int X)** retournant **1** si **X** est **impair** et **0** sinon.
b. Afficher un message à l'utilisateur, lui demandant de donner un nombre entier **X**.
c. Appeler la fonction **impair(X)**
d. Afficher un **1** si **X** est **impair** et **0** si **X** est **pair** en appelant une fonction **print_int**.