University of 08 Mai 1945 – Guelma

Faculty of MISM | **Date: 04/06/2024**
Department of Mathematics | **Duration : 1h30**
First year LMD | **Semester 2**

Resit Exam of Algorithmics and Data Structures 2

In all exercises, the solution should be presented in algorithmic form rather than in the C language.

في جميع التمارين، يجب أن تكون الحلول في شكل خوارزميات وليس في شكل برامج باللغة C.

Dans tous les exercices, la solution doit être présentée sous forme algorithmique plutôt qu'en langage C.

Exercise 1 (5.5 pts):

1) Write the **iterative** function **`nbMultIt`** that takes a positive integer **`n`** as input and calculates and returns the number of values between 1 and **`n`** that are multiples of 2 but not multiples of 5.
2) Write the **recursive** function **`nbMultRec`** that takes a positive integer **`n`** as input and calculates and returns the number of values between 1 and **`n`** that are multiples of 2 but not multiples of 5.

---

1) اكتب الدالة التكرارية **`nbMultIt`** التي تأخذ عدد صحيح موجب **`n`** كمدخل، وتقوم بحساب وإرجاع عدد القيم المحصورة بين 1 و **`n`** التي هي من مضاعفات 2 ولكن ليست من مضاعفات 5.
2) اكتب الدالة التراجعية **`productRec`** التي تأخذ عدد صحيح موجب **`n`** كمدخل، وتقوم بحساب وإرجاع عدد القيم المحصورة بين 1 و **`n`** التي هي من مضاعفات 2 ولكن ليست من مضاعفات 5.

---

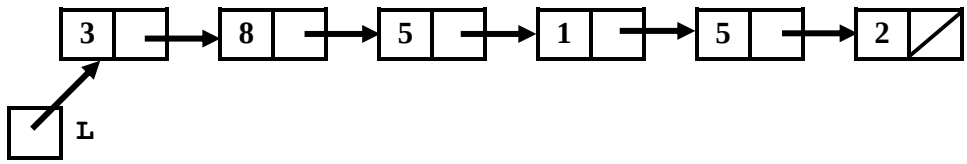
1) Écrire la fonction **itérative** **`nbMultIt`** qui prend un entier positif **`n`** en entrée et calcule et renvoie le nombre de valeurs comprises entre 1 et **`n`** qui sont des multiples de 2 mais pas des multiples de 5.
2) Écrire la fonction **récursive** **`nbMultRec`** qui prend un entier positif **`n`** en entrée et calcule et renvoie le nombre de valeurs comprises entre 1 et **`n`** qui sont des multiples de 2 mais pas des multiples de 5.

Exercise 2 (6.5 pts):

Write the function **`sumFromKth`** that takes a linked list of integers **`L`** and an integer **`k`** as input, and that calculates and returns the sum of the elements of **`L`** starting from the **`k`**th element onwards. If the list does not contain enough elements to reach the **`k`**th element, the function should return 0.

Note: Assume that the first element of the list is at position 1.

Example: If the list **`L`** is as follows:



- For **`k`** = 4, the function should return the value **8**, which is the sum of 1 + 5 + 2.
- For **`k`** = 7, the function should return the value **0**.

اكتب الدالة **`sumFromKth`** التي تأخذ كمدخل قائمة مترابطة من الأعداد الصحيحة **`L`** وعدد صحيح **`k`** والتي تقوم بحساب وإرجاع مجموع عناصر **`L`** بدءًا من العنصر رقم **`k`**. إذا لم تحتوي القائمة على عدد كافٍ من العناصر للوصول إلى العنصر رقم **`k`**، يجب أن تعيد الدالة القيمة 0.

ملاحظة : اعتبر أن العنصر الأول في القائمة يكون رقمه 1.

مثال: إذا كانت القائمة المترابطة كما هو موضح في الشكل السابق:

- من أجل **k = 4**، يجب أن تعيد الدالة القيمة **8**، والتي هي مجموع 1 + 5 + 2.
- من أجل **k = 7**، يجب أن تعيد الدالة القيمة **0**.

Écrire la fonction **sumFromKth** qui prend en entrée une liste chaînée d'entiers **L** et un entier **k**, et qui calcule et retourne la somme des éléments de **L** à partir du **k**ième élément inclus. Si la liste ne contient pas suffisamment d'éléments pour atteindre le **k**ième élément, la fonction doit retourner 0.

Remarque **:** Considérez que le premier élément de la liste est à la position 1.

Exemple **:** Si la liste **L** est comme illustré par la figure précédente :

- Pour **k** = 4, la fonction doit retourner la valeur **8**, qui est la somme de 1 + 5 + 2.
- Pour **k** = 7, la fonction doit retourner la valeur **0**.

Exercise 3 (8 pts):

Let **S** be a non-empty stack of integers:

1) Write the function **average** that calculates and returns the average of the elements in the stack **S** passed as a parameter.
2) Using the function **average**, write the procedure **split** that takes as input a stack of integers **S** and produces as output two other stacks, **S1** and **S2**. **S1** should contain the elements of **S** that are strictly less than the average of the stack, and **S2** should contain the elements of **S** that are greater than or equal to the average of the stack. The original stack should remain unchanged.

ليكن **S** مكدسًا (Stack) غير فارغ من الأعداد الصّحيحة:

1) اكتب الدّالة **average** التي تحسب وتعيد متوسّط عناصر المكدس **S** التي تم تمريره كمدخل.
2) باستخدام الدّالة **average** ، اكتب الإجراء **split** الذي يأخذ كمدخل مكدساً من الأعداد الصحيحة **S** وينتج كمخرج مكدسين آخرين **S1** و **S2**. يجب أن يحتوي **S1** على عناصر **S** التي هي أقل تماما من متوسط المكدس، و**S2** على عناصر **S** التي هي أكبر من أو تساوي متوسط المكدس. يجب أن يبقى المكدس الأصلي بدون تغيير.

Soit **S** une pile non vide d'entiers:

1) Ecrire la fonction **average** qui calcule et renvoie la moyenne des éléments de la pile **S** passée comme paramètre.
2) En utilisant la fonction **average**, écrire la procédure **split** qui prend en entrée une pile d'entiers **S**, et qui produit en résultat deux autres piles **S1** et **S2**. **S1** doit contenir les éléments de **S** qui sont strictement inférieurs à la moyenne de la pile, et **S2** doit contenir les éléments de **S** qui sont supérieurs ou égaux à la moyenne de la pile. La pile originale doit rester inchangée.

Good luck

Dr. Abderrahmane Kefali

« Our greatest weakness lies in giving up. The most certain way to succeed is always to try just one more time. »

[Thomas A. Edison]