

LABORATOIRE 6

But : Exercer la programmation structurée, le dimensionnement de tableau à l'exécution et analyser la complexité d'algorithmes

Donnée :

Dans un premier temps, définir un type `FLOTTANT_ETENDU` permettant de représenter des nombres en virgule flottante dont l'étendue des exposants est bien supérieure à celle des `double`. Pour cela, on mémorise la mantisse avec un `double` compris, en valeur absolue, entre 0.1 et 1 (ou 0, pour représenter le 0) et l'exposant au moyen d'un `int`.

Définir des fonctions permettant de transformer un `int` en `FLOTTANT_ETENDU`, de multiplier deux `FLOTTANT_ETENDU` et d'afficher un nombre de ce type.

Tester ces fonctions en écrivant un programme qui demande un nombre entier n non négatif à l'utilisateur et qui affiche la valeur de $n!$

Dans un deuxième temps, écrire un programme qui affiche la valeur de $n!$ avec tous les chiffres. Pour cela, déclarer un tableau d'entiers dont la taille sera celle donnée par l'exposant du

`FLOTTANT_ETENDU` représentant $n!$

La $j^{\text{ème}}$ composante de ce tableau contiendra finalement le $j^{\text{ème}}$ chiffre de $n!$

Ce tableau sera initialisé à 0 0 0 0 ... 0 1 (qui est la valeur de $0!$)

À la $j^{\text{ème}}$ étape, on commence par multiplier toutes les composantes de ce tableau par i , puis on fait passer les « retenues » vers la composante de poids plus élevé, si une valeur dépasse 9.

Estimer, en fonction de n , l'ordre de grandeur (linéaire, quadratique, cubique, ...) de la croissance du nombre d'opérations que doit réaliser le programme pour calculer la valeur approchée de $n!$ au moyen de `FLOTTANT_ETENDU`, et pour calculer la valeur exacte de $n!$

Durée : 2 séances