

Auteur : Raymond Moché

Nous donnons deux solutions au problème posé : « AF_Som_carres_expurge.txt » et « AF_Som_carres_expurge.txt » qui va un peu plus loin en permettant de choisir le nombre maximum de carrés que l'on somme (appelé n dans le script, grâce à la commande « input » et qui soigne l'affichage). Rappelons que les fichiers *scilab* (extensions sce ou sci) sont des fichiers texte ordinaires (extension txt).

Commentaires sur le premier fichier : « AF_Som_carres_expurge.txt » rebaptisé « AF_Somcarexp.sce »

```
Startup execution:
  loading initial environment

-->exec("/Users/raymondmoche/Magasin_scilab/AF_Som_carres_expurge.txt")

-->// Somme des carres des n premiers entiers.

-->function [x,y] = f(n)
-->  x = n*(n+1)*(2*n+1)/6;
-->  y = 0;
-->  for j = 1:n
-->    y = y+j^2;
-->  end
-->endfunction

-->// le nom de cette fonction est "f".

-->//Passons au comptage du nombre d'egalite (E)

-->// non verifiees jusqu'a 500

-->S=0;

-->for j=1:500
-->  [a,b]=f(j);
-->  if a<>b then
-->    S=S+1;
-->  end
-->end

-->// Somme des 500 premiers carres :

-->a
a =

  41791750.

-->// Nombre d'egalites non satisfaite :

-->S
S =

  0.
```

On appelle ce fichier à l'aide de la commande `exec("chemin et nom du fichier")`, voir ligne 3 de la capture d'écran. *scilab* recopie le script et l'exécute. Comme les commandes a et S n'ont pas été terminées par le signe « ; », les valeurs de a et S sont affichées (sinon, elles auraient été calculées sans être affichées). On lit ainsi que $1^2 + 2^2 + \dots + 500^2 = 41751750$ et qu'il n'y a pas d'égalité de la forme

$$1^2 + \dots + j^2 = \frac{j(j+1)(2j+1)}{6}$$

qui ne soit pas vérifiée quand j varie de 1 à 500. On ne peut rien en déduire quand $j > 500$.

Voir ci-contre les commentaires sur le deuxième fichier

Commentaires sur le deuxième fichier : « AF_Som_carres.txt », rebaptisé « AF_Somcar.sce »

```
-->exec("/Users/raymondmoche/Magasin_scilab/AF_Som_carres.txt")
```

```
-->// Somme des carres des n premiers entiers.
```

```
-->function [x,y] = f(n)
--> x = n*(n+1)*(2*n+1)/6;
--> y = 0;
--> for j = 1:n
-->   y = y+j^2;
--> end
-->endfunction
```

```
-->// le nom de cette fonction est "f".
```

```
-->//Passons au comptage du nombre d'egalite (E)
```

```
-->// non verifiees jusqu'a n
```

```
-->n=input("Quelle valeur pour n?")
Quelle valeur pour n?500
```

```
n =
    500.
```

```
-->S=0;
```

```
-->for j=1:n
--> [a,b]=f(j);
--> if a<>b then
-->   S=S+1;
--> end
-->end
```

```
-->S
S =
    0.
```

```
-->// Affichage du resultat
```

```
-->if (S>0) then
--> a="(E) pas toujours verifiee";
-->else
--> a="(E) est verifiee de 1 a n";
-->end
```

```
-->a
a =
(E) est verifiee de 1 a n
```

```
-->n
n =
    500.
```

```
-->// Somme des n premiers carres
```

```
-->b
b =
41791750.
```

Ce fichier est plus intéressant que le précédent car il est maintenant permis de choisir n : *scilab* charge et exécute le script comme précédemment. Quand il arrive à la commande

```
n=input("Quelle valeur pour n?")
```

il pose la question. Il suffit de taper 500 si l'on choisit $n=500$ (taper 5000, par exemple, montrerait que le temps de calcul n'est plus négligeable). L'exécution de l'algorithme reprend. S va s'afficher parce que cela a été demandé. Ensuite, comme il y a *a priori* deux affichages possibles, on choisit la bonne réponse à l'aide d'une *instruction conditionnelle avec alternative*. Remarquons que l'instruction

```
a="(E) pas toujours verifiee";
```

définit a comme une chaîne de caractères dont la syntaxe est contraignante (pas d'apostrophe par exemple). Finalement, on demande que cette réponse soit affichée (instruction a) ainsi que n (que l'on pourrait avoir oublié) et b (qui est $1^2 + \dots + n^2$).