

Enoncé "fonction mystérieuse" :

Le but de cet exercice est de fournir un équivalent C de la fonction assembleur suivante :

```
1      _foncmyst:
2          li r2,1
3          li r9,0
4          cmpw cr7,r2,r3
5          bgt- cr7,L8
6      L6:
7          divw r0,r3,r2
8          mullw r0,r0,r2
9          subf r0,r0,r3
10         add r9,r9,r0
11         addi r2,r2,1
12         cmpw cr7,r2,r3
13         ble+ cr7,L6
14     L8:
15         mr r3,r9
16         blr
```

1. Isoler et désigner précisément les instructions permettant de calculer un modulo (en C, la fonction " reste " ou " modulo " s'écrit " % "). Justifier.
2. Combien de paramètres comporte cette fonction ? Quel est leur type ? Justifier.
3. Repérer précisément la boucle. Quel registre contrôle son déroulement ? Comment est-il initialisé ? Quelle est la condition d'arrêt ? Comment ce registre évolue-t-il (progression)?
4. La boucle utilise-t-elle des registres de travail ? Si oui, lesquels et pourquoi ?
5. De quel type est le résultat fourni par la fonction et comment est-il calculé ?
6. Donner maintenant un équivalent C de la fonction mystérieuse fournie ci-dessus.

==== fin troisième série d'exercices =====