

Responsable pédagogique

Période

Volume horaire

AF	AM	PB	PM
Sem1	Sem2	Sem3	Sem4
Cours/TD		TP	
		6	

STRUF TP Conversions Romain

Indicateur temporel

questions	1h	2h	3h	4h	5h	6h
1						
2						
3-4						

Tous les fichiers de ce TP se trouvent dans le répertoire /pub/STRUF de la machine OUVEA (dans votre voisinage réseau).

Partie 1 : conversion en nombre romain ancien

Le but est de coder un programme effectuant une conversion de nombre décimal arabe en romain ancien. Le système de numération Romain utilise les symboles M, D, C, L, X, V et I ; associées respectivement , en base décimale, à : 1000, 500, 100, 50, 10, 5, 1.

Un nombre se lit de gauche à droite, sa valeur est la somme des valeurs décimales associées aux symboles.

Principe de l'algorithme

Le principe consiste à retirer du nombre initial, et ce à chaque itération, la valeur décimale équivalente au plus grand symbole contenu dans ce nombre. On affiche bien sûr le symbole associé, à chaque fois que l'on retranche cette valeur décimale.

(Ex : $1320 > 1000$ donc affiche M et nouveau reste = $1320 - 1000 = 320$).

$1320 = 1000 + 100 + 100 + 100 + 10 + 10 \Rightarrow$ affichage de 'M' 'C' 'C' 'C' 'X' 'X'

1. Saisie et vérification de l'entier à convertir

A partir du source dec2rom.c fourni, on demande de coder la saisie sur la ligne de commande , du nombre entier à convertir.

Pensez à utiliser la macro-fonction *IntegerArg(N)* , et tester que le nombre N est bien compris entre 1 et 3999.

Si votre nombre n'est pas dans cet intervalle, sortez du programme par return 0, sinon vous devez l'afficher simplement en " echo ". (on ne fait pas encore la conversion ici)

Compilez en faisant `gcc -Wall -o dec2rom dec2rom.c`
Lancer votre exécutable `./dec2rom`

Qu'observez-vous dans le terminal ? Notez au moins 2 exemples montrant que votre nombre est correctement saisi.

.....

.....

.....

2. Codage de la boucle de conversion

On dispose de 2 tableaux ; Symb_romain contenant les caractères liés à la base de numération romaine, et Symb_arabe contenant les valeurs décimales équivalentes.

```
const char *Symb_romain = {"MDCLXVI"};
```

```
const int Symb_arabe[] = {1000,500,100,50,10,5,1};
```

On demande de transcrire en C, l'algorithme donné ci-dessous.

Remarque :

Les accolades correspondent à des commentaires en pseudo-langage et se traduisent en C par /* pour { et */ pour }

La flèche d'affectation, se traduit par un = en C.

Algorithme proposé :

reste ← nb ; { on met le nombre saisi, dans une variable reste }

TANT QUE (reste > 0) FAIRE

{ Reste > nombre arabe associé à indice dans le tableau }

SI (Reste >= Symb_arabe[indice]) ALORS

Afficher(Symb_romain[indice]); { Afficher symbole romain associé }

Reste ← reste - Symb_arabe[indice]; { Mise à jour du reste du nombre }

SINON {Autrement on passe au nb arabe suivant du tableau }

Indice ← indice + 1 ;

FINSI

FINTQ

Compilez en faisant gcc -Wall -o dec2rom dec2rom.c

Lancer votre exécutable ./dec2rom

Qu'observez-vous ? Donnez des exemples

.....
.....
.....
.....
.....
.....

3. Influence du test de la boucle

Modifier le <= en < dans le test de boucle de conversion.

Compilez en faisant gcc -Wall -o dec2rom dec2rom.c

Lancer votre exécutable ./dec2rom

Qu'observez-vous ?

.....
.....
.....



.....

Avez-vous une explication ?

.....

.....

.....

Partie 2 : conversion en romain moderne

Le principe est le même que pour le romain ancien, avec simplement le fait que pour les nombres 900, 400, 90, 40, 9 et 4, on associera respectivement " CM ", " CD ", " XC ", " XL ", " IX ", " IV ".

4. Codage de la solution

Faites une copie de votre source `dec2rom.c` et nommez-le *newdec2rom.c*.

Vous ne travaillerez que sur ce fichier *newdec2rom.c*.

Modifier vos 2 tableaux `Symb_arabe` et `Symb_romain`, afin d'y rajouter les nombres et chaînes de caractères précisés ci-dessus.

Compilez, tester.

Résultats obtenus :

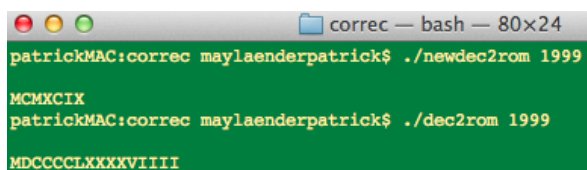
.....

.....

.....

.....

Annexe : exemple d'exécution des 2 conversions



```
correc — bash — 80x24
patrickMAC:correc maylaenderpatrick$ ./newdec2rom 1999
MCMXCIX
patrickMAC:correc maylaenderpatrick$ ./dec2rom 1999
MDCCLXXXV
```