PARCOURS 6 :

Dans la mesure du possible privilégier l'utilisation des fonctions pour résoudre les exercices ci-dessous.

EXERCICE 33 : évolution population

1°) Dans un pays de population constante égale à 60 millions d'habitants, on compte 20 millions de citadins et 40 millions de ruraux en 2005. Les habitants vivent soit en zone rurale, soit en ville et on constate que les mouvements de population suivent la règle suivante : chaque année, 10% des citadins émigrent en zone rurale et 20% des ruraux émigrent en zone urbaine.

a) Écrire une fonction permettant de calculer la population dans chaque zone après n années.

b) Quelle évolution peut-on prévoir à long terme ?

2°) *Cas général* : Dans un pays de population globale constante, les habitants vivent soit en zone rurale, soit en ville et on constate que des ruraux émigrent en zone urbaine et des citadins émigrent en zone rurale.

a) Écrire une fonction qui retourne la population dans chaque zone après n années, et qui prend comme arguments le nombre n, le nombre initial c0 de millions de citadins, le nombre initial t0 de millions de ruraux, la proportion tc % (supposée constante) des citadins qui émigrent en zone rurale annuellement et la proportion tr % (supposée constante) des ruraux qui émigrent en zone urbaine chaque année.

b) Quelle évolution peut-on prévoir en 2030 dans un pays comptant, en 2017, 900 millions d'habitants dont 400 millions de citadins, dans lequel, chaque année, 20% des ruraux émigrent en zone urbaine et 5% des citadins émigrent en zone rurale ?

EXERCICE 34 :

Écrire une fonction quotient et une fonction reste qui renvoient le quotient et le reste de la division euclidienne. Ces fonctions devront être écrites en n' utilisant que des additions et des soustractions, les fonctions // et % ne pourront pas être utilisées.

EXERCICE 35 :

© Nicolas Pazat

Si a et b sont deux entiers naturels, on rappelle que le pgcd(a,b) (le plus grand commun diviseur de a et de b) peut se calculer par l'algorithme d'Euclide :

a) Si b = 0 alors pgcd(a,b) = a.
b) Sinon, on calcule le reste de la division euclidienne de a par b.
   On utilise que pgcd(a,b) = pgcd(b,r) et on remplace la valeur de a par celle de b, la valeur de b par celle de r.

On itère le procédé jusqu'à trouver b = 0. La valeur de pgcd(a,b) est alors contenue dans a.

a) On prend a = 192 et b = 138.
   Faites fonctionner « à la main » l'algorithme d'Euclide en complétant le tableau suivant :

a	b	r
192	138	54
138	54	

   Préciser la valeur de pgcd(192,138) : …………………………………

b) Écrire un programme saisissant au clavier deux entiers naturels a et b et affichant à l'écran pgcd(a ;b).
   Utiliser ce programme pour calculer pgcd(51940,6201) : …………………………………

EXERCICE 36 :

© Nicolas Pazat

On se donne un entier naturel n. Si n s'écrit avec plusieurs chiffres, alors on calcule le produit de ses chiffres ce qui nous donne un nouvel entier naturel. On recommence ce procédé sur ce nouvel entier et ainsi de suite jusqu'à obtenir un seul chiffre.

Par exemple : 377 → 147 → 28 → 16 → 6

On appelle persistance d'un nombre le nombre d'étapes nécessaires pour aller du nombre n à un seul chiffre, pour 377 la persistance est de 4.

1°) Écrire une fonction persistance(n), qui connaissant un entier naturel n, donne sa persistance.

2°) Écrire une fonction minPer(p) qui renvoie le plus petit entier naturel n qui a pour persistance p.

On trouvera minPer(0) = 0, minPer(1) = 10, minPer(2) = 25 …

Utiliser la fonction minPer pour compléter le tableau ci-dessous :

p	5	7	8
minPer(p)			

REMARQUE :

À l'aide d'ordinateurs puissants des chercheurs ont pu calculer minPer(11) qui vaut 277777788888899 et on pense actuellement qu' aucun entier n'a une persistance strictement supérieure à 11 (mais ce n'est pas encore démontré).

En savoir plus : Pour la Science n° 430 août 2013

EXERCICE 37 :

© Nicolas Pazat

L'ordinateur tire un nombre entier au hasard entre 0 et 200. L'utilisateur doit proposer des entiers pour trouver le bon en 10 coups maximum. À chaque proposition, l'ordinateur doit afficher le numéro de l'essai et si le nombre est trop petit, trop grand, ou si c'est le bon, auquel cas l'ordinateur le félicite et lui dit en combien de coups il l'a trouvé.

1°) Écrire ce programme.

2°) Modifier le programme pour que l'ordinateur demande après chaque partie si l'utilisateur veut en refaire une. Lorsque la réponse est non, le script doit afficher le nombre de parties effectuées, le nombre de victoires et le nombre moyen de coups qui ont été nécessaires pour trouver s'il y a eu au moins une victoire.

EXERCICE 38 : Multiplication Egyptienne/Russe

La multiplication russe est une technique qui permet de multiplier deux nombres entiers A et B en ne faisant que :

- *des multiplications par 2*
- *des divisions par 2*
- *des additions*

Exemple 1 : Calculer le produit de 35 par 32

A	B
35	32
70	16
140	8
280	4
560	2
1120	1

La propriété d'associativité de la multiplication suffit à justifier ce calcul : 35 x 32 = 35 x (2 x 16) = (35 x 2) x 16 = 70 x 16 = … = 1120

Dans cet exemple, le facteur B étant une puissance de 2, des multiplications par 2 et des divisions par 2 suffisent à calculer le produit A x B

Exemple 2 : Calculer le produit de 47 par 38

Ici 19 = 2x9 + 1 donc

94 x 19 = 94 x (2x9+1) = 94 x 2 x 9 + 94 x 1 = 188 x 9 + 94

De même : 9 = 2x4 + 1 donc

188 x 9 = 188 x (2x4 + 1) = 376 x 4 + 188

Donc 47 x 38 = 1504 + 188 + 94 = 1786

A	B	« reste »
47	38	
94	19	
188	9	94
376	4	188
752	2	
1504	1	

Proposer un algorithme ou même une fonction « multiplication_russe (A,B) » qui calcule le produit de A par B par cette technique.