

REIMS

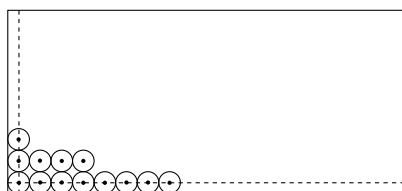
Exercice n° 1 (Toutes Séries)

Enoncé

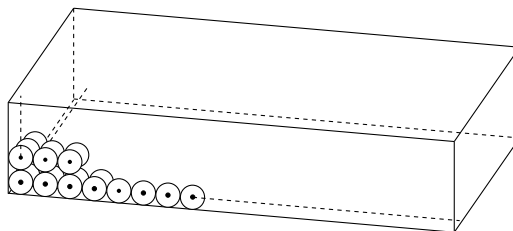
Remplissage d'une boîte par des billes

On dispose d'une boîte de dimensions : $L = 10$ cm, $\ell = 5$ cm et $h = 2$ cm.

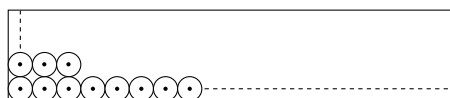
On y range des billes de diamètre d (d étant un nombre entier de millimètres) de la façon suivante :



vue de dessus



perspective



vue de face

(Les centres des billes sont alignés parallèlement aux arêtes de la boîte.)

1. Chercher le ou les rayons des billes leur permettant d'occuper un volume maximal
2. Donner ce volume maximal.

Eléments de solution

d étant le diamètre d'une bille en millimètres et $E(x)$ désignant la partie entière de x .

Nombre de billes sur la longueur : $E\left(\frac{100}{d}\right)$

Nombre de billes sur la largeur : $E\left(\frac{50}{d}\right)$

Nombre de billes sur la hauteur : $E\left(\frac{20}{d}\right)$

Volume :

$$E\left(\frac{100}{d}\right) \times E\left(\frac{50}{d}\right) \times E\left(\frac{20}{d}\right) \times \frac{4}{3}\pi$$

$$\leq \left(\frac{100}{d}\right) \times \left(\frac{50}{d}\right) \times \left(\frac{20}{d}\right) \times \frac{4}{3}\pi \left(\frac{d}{2}\right)^3 = \frac{50000}{3}\pi \text{ mm}^3$$

Le volume maximal est $\frac{50000}{3}\pi \text{ mm}^3$

C'est-à-dire pour $d = 1 \text{ mm}$, 2 mm , 5 mm ou 10 mm .

Donc pour $r = 0,5 \text{ mm}$, 1 mm , $2,5 \text{ mm}$ ou 5 mm .

Exercice n° 2 (Séries autres que S)

Enoncé

Le code ASCII

Écriture binaire :

En informatique, on utilise la propriété suivante :

Tout nombre entier strictement inférieur à 256 s'écrit de manière unique sous la forme $a \times 2^7 + b \times 2^6 + c \times 2^5 + d \times 2^4 + e \times 2^3 + f \times 2^2 + g \times 2^1 + h$ où a, b, c, d, e, f et g sont égaux à 0 ou 1.

Le clavier standard d'un vieux ordinateur des années 80 correspond aux caractères ASCII.

Les codes 65, 66, ... 90 dans cet ordre permettent de décrire les lettres de l'alphabet de A à Z.

Exemples :

Code ASCII	Positions mémoires	Forme	Valeurs
		128 64 32 16 8 4 2 1	
65	46600		8
	46601		20
	46602		34
	46603		34
	46604		62
	46605		34
	46606		34
	46607		0

Code ASCII	Positions mémoires	Forme								Valeurs
66		128	64	32	16	8	4	2	1	
	46608									60
	46609									34
	46610									34
	46611									60
	46612									34
	46613									34
	46614									60
	46615									0

Toutes ces lettres conservent les deux colonnes de gauche blanches pour séparer les lettres d'un mot.

1. Compléter dans le tableau suivant, qui correspond à la lettre Z, les positions mémoire et les valeurs :

Code ASCII	Positions mémoires	Forme								Valeurs
90		128	64	32	16	8	4	2	1	
										
										
										
										
										
										
										

2. Un caractère peut être complètement ou partiellement redéfini pour créer une autre police, de nouveaux symboles...

Pour modifier le contenu d'une position mémoire en programmation BASIC, on se sert de la commande POKE m, a qui, à la position mémoire, m place la valeur a .

On transforme la lettre A en écrivant :

POKE 46600,28

POKE 46601,34

POKE 46602,32

POKE 46603,60

POKE 46604,34

POKE 46605,34

POKE 46606,28

POKE 46607,0

Dessiner le nouveau caractère.

3. Dans le cas d'une lettre, qu'obtient-on si, en utilisant la commande POKE m, a on multiplie toutes ses valeurs par 2 ?
4. Quelle opération doit-on effectuer sur les valeurs d'un caractère pour avoir le même caractère en blanc sur fond noir plutôt qu'en noir sur fond blanc ?

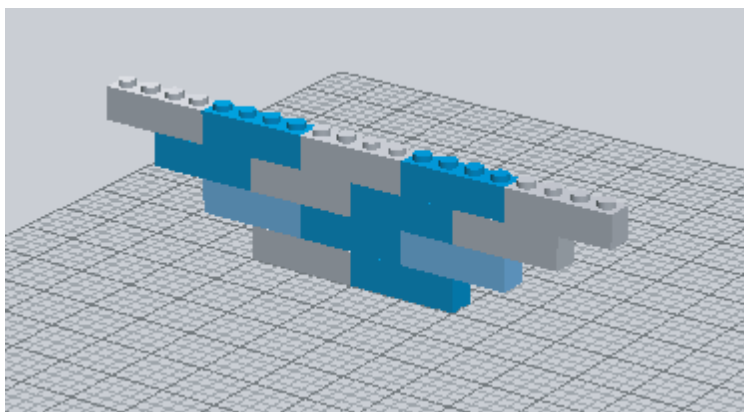
Exercice n° 3 (Série S)

Énoncé

Un jeu de construction

Arthur joue avec son baril de petites briques emboîtables. Elles sont toutes identiques et il a décidé de les empiler de la façon suivante :

Il commence par placer une rangée de n briques sur le sol. Dessus, il en place une rangée qui débordé également de chaque côté et contient une brique de plus et ainsi de suite. A chaque fois, il essaye d'obtenir une configuration utilisant toutes les briques qu'il a choisies et comportant au moins deux rangées. Par exemple, pour empiler 15 briques, il en positionne 7 sur la première rangée puis 8 sur la deuxième. Pour en utiliser 14, il commence avec une rangée à 2, puis 3, 4 et enfin 5.



1. Il essaye en utilisant exactement 100 briques et réussit. Combien en a-t-il mis sur la première rangée ? On demande toutes les possibilités.
2. Il en utilise exactement N . Montrer que si N peut se mettre sous la forme $N = (2p+1)q$, avec p et q deux entiers non nuls, alors Arthur peut réussir.
3. Il décide d'utiliser l'intégralité du stock de briques contenues dans son baril. Et là, malgré ses diverses tentatives et sa ténacité, il n'arrivera jamais à compléter le dernier rang. Le baril affiche « entre 500 et 600 pièces ». Combien en a-t-il exactement ?
4. Quels sont les nombres N pour lesquels il n'y a pas de solution possible ?

Rappel : $1 + 2 + 3 + \dots + n = \frac{n(n+1)}{2}$.

Eléments de solution

1. En essayant toutes les décompositions, on trouve seulement deux possibilités : $100 = 18 + 19 + 20 + 21 + 22 = 10 + 11 + 12 + 13 + 14 + 15 + 16$

1. $n + (n + 1) + \dots + (n + b) = N$

donc $n(b + 1) + \frac{n(b + 1)}{2} = N$

soit : $\frac{(b + 1)(2n + b)}{2} = N$.

Avec une calculatrice par A. Guillemot

Soi n le nombre de briques dans la première rangée et soit p le nombre de rangées.

On a donc la relation :

$$n + (n + 1) + (n + 2) + \dots + (n + p - 1) = 100$$

C'est-à-dire $n \times p + (1 + 2 + \dots + p - 1) = 100$.

Qui peut s'écrire $n \times p + \frac{(p - 1) \times p}{2} = 100$.

Exprimons n en fonction de p

$$n = \frac{100}{p} - \frac{p - 1}{2}.$$

Comme $1 + 2 + 3 + \dots + 15 = 105$, il ne peut pas avoir plus de 14 rangées de briques pour cette question.

Examinons la table de valeurs de la fonction définie par $Y_1 = \frac{100}{x} - \frac{(x - 1)}{2}$ pour les entiers entre 1 et 14.

Pour répondre au problème, Y_1 devra être un entier.

X	Y ₁	
1	100	
2	49.5	
3	32.333	
4	23.5	
5	18	
6	14.167	
7	11.286	
X=1		

X	Y ₁	
8	9	
9	7.1111	
10	5.5	
11	4.0909	
12	2.8333	
13	1.6923	
14	.64286	
X=14		

Nous constatons qu'il y a deux solutions au problème :

5 rangées : 18, 19, 20, 21, 22 ;

8 rangées : 9, 10, 11, 12, 13, 14, 15, 16.