

be-OI 2026	Remplissez ce cadre en MAJUSCULES et LISIBLEMENT svp	O
Finale - CADET samedi 14 mars 2026	PRÉNOM : NOM : ÉCOLE :	Réservé

Finale de l'Olympiade belge d'Informatique 2026 (durée : 2h)

Consignes à lire attentivement avant l'épreuve.

1. **Attendez le signal du départ** avant d'enlever les feuilles de la pochette en plastique.
2. Vous devez passer la finale dans la même langue que celle dans laquelle vous avez passé la qualification.
Vérifiez que cette page est dans la bonne langue. Si ce n'est pas le cas, signalez-le ou changez de place.
3. Vérifiez que vous avez reçu la **bonne série de questions** mentionnée ci-dessus dans l'en-tête.
 - Pour les élèves jusqu'en deuxième année du secondaire: catégorie **cadet**, feuilles **roses**.
 - Pour les élèves en troisième ou quatrième année du secondaire: catégorie **junior**, feuilles **vertes**.
 - Pour les élèves de cinquième année du secondaire et plus: catégorie **senior**, feuilles **jaunes**.
4. Quand le signal de départ est donné, sortez les feuilles de la pochette, mais **n'enlevez pas les agrafes**.
5. Indiquez votre nom, prénom et école **très lisiblement en MAJUSCULES et uniquement sur cette page**.
6. Indiquez vos **réponses sur les feuilles de couleur** agrafées avec cette page.
Écrivez de façon **bien lisible** à l'aide d'un **bic ou stylo** bleu ou noir.
7. Utilisez les feuilles blanches comme feuilles de brouillon.
N'oubliez pas de recopier vos solutions sur les feuilles de couleur.
8. L'épreuve dure 2 heures et vous devez rester sur place jusqu'à la fin.
Si vous avez terminé plus tôt, appelez un surveillant pour remettre vos réponses.
9. Quand le signal de fin est donné vous devez immédiatement remettre vos solutions.
 - Ne remplacez pas les feuilles dans la pochette plastique.
 - **Déposez les feuilles de couleur, toujours agrafées**, dans une des boîtes prévues à cet effet.
 - **Rendez la pochette en plastique** en la déposant dans une boîte prévue à cet effet.
 - Gardez les feuilles blanches.
10. Vous ne pouvez avoir que de quoi écrire avec vous. Les calculatrices, smartphones, ... sont **interdits**.
11. Tous les extraits de code des énoncés sont en **pseudo-code**. Vous trouverez, sur les pages suivantes, une **description** du pseudo-code que nous utilisons. Si vous devez répondre en code, vous pouvez utiliser le **pseudo-code** ou un **langage de programmation courant** (Java, C, C++, Pascal, Python, ...). Les erreurs de syntaxe ne sont pas prises en compte pour l'évaluation.

L'Olympiade Belge d'Informatique est possible grâce au soutien de nos membres:





Aide-mémoire pseudo-code

Les données sont stockées dans des variables, chacune est identifiée par un nom.

Le nom permet d'accéder à la variable pour placer ou récupérer les données.

On utilise \leftarrow comme opérateur d'affectation pour stocker une valeur dans une variable.

Exemple : $n \leftarrow 100$ place le nombre entier 100 dans la variable dénommée n.

Variables et données simples	Exemples
nombre entier	$n \leftarrow 100$ $m \leftarrow -1$
nombre réel	$pi \leftarrow 3.1415$ $z \leftarrow 0.0$
valeur logique (booléen)	$t \leftarrow \mathbf{true}$ (vrai) $t \leftarrow \mathbf{false}$ (faux)

On peut effectuer des opérations arithmétiques en utilisant des nombres et des variables.

addition et soustraction avec + et - (exemple : $a+3$)
multiplication avec * ou \times (exemple : $2*a*b$)
division entière : quotient avec / ou // et reste avec % (exemple : $14//3$ est égal à 4 et $14\%3$ est égal à 2)
division non entière / : peu utilisée, on signale si c'est le cas (exemple : $3.6/4.0$ est égal à 0.9)
puissance ou exposant avec ^ (exemple : x^3 est égal à $x*x*x$)

On utilise souvent des variables pour calculer un résultat et le placer ensuite dans une variable.

Parfois le résultat est placé dans une des variables utilisées dans le calcul.

Le cadre à droite contient du code qui illustre cela.

Après l'exécution de ce code, $x=25$, $a=4$ et $b=10$.

```
a ← 3
b ← 5
x ← a*b + 10
a ← a + 1
b ← b*2
```

```
if (a < b)
    { p ← p+5 }
else
    { p ← p-2 }
c ← c-1
```

L'instruction **if** permet d'exécuter du code uniquement si une condition est vraie. On peut éventuellement ajouter l'instruction **else** pour exécuter un autre code uniquement si la condition est fausse.

Dans l'exemple à gauche, si le nombre dans la variable a est plus petit que celui dans la variable b, alors on ajoute 5 dans la variable p, sinon on soustrait 2 dans la variable p.

Ensuite, dans tous les cas, on enlève 1 dans la variable c.

Les instructions à exécuter dans les différents cas seront clairement identifiées soit en les plaçant entre accolades { }, soit en les décalant.

Bien souvent, on fera les deux : mise entre accolades et décalage, comme ci-dessus.

Voici la liste des opérateurs de comparaison les plus courants.

= ou == est égal à	< plus petit que	<= ou ≤ plus petit ou égal	> plus grand que	>= ou ≥ plus grand ou égal	!= ou ≠ est différent de
-----------------------	---------------------	-------------------------------	---------------------	-------------------------------	-----------------------------

On peut tester plusieurs conditions en les combinant avec les opérateurs logiques **and**, **or**, **not**.

La condition $(p \text{ and } q)$ est vraie si les 2 conditions p et q sont vraies.

La condition $(p \text{ or } q)$ est vraie si au moins une des 2 conditions est vraie.

La condition $\text{not}(p)$ est vraie si p est fausse et fausse si p est vraie.

On peut placer la valeur logique d'une condition dans une variable booléenne à utiliser plus tard.

Exemple : $f \leftarrow ((a==5) \text{ and } (b>=0)) \text{ or } ((a<0) \text{ and } (b!=10))$

La condition d'un **if** est parfois une simple variable booléenne.

Exemple : **if** (f) {a←10} **else** {b←10}

Il faut parfois stocker plusieurs données dans une seule variable structurée.

Variabes et données structurées	Exemples
tableau, liste, vecteur	$seq \leftarrow [7, 11, 0, -4, 9]$
tableau, matrice	$M \leftarrow [[0, 1, 2], [3, -1, 3], [0, 0, 5]]$
couple	$coord \leftarrow (1, 7)$
texte	$n \leftarrow \text{"John"}$

Les éléments individuels d'une variable structurée sont repérés par un indice que l'on écrit entre crochets après le nom de la variable.

Le premier élément d'une variable structurée G est d'indice 0 et est noté $G[0]$.

Le second élément est d'indice 1 et le dernier est d'indice $n-1$ s'il y a en tout n éléments.

Le nombre d'éléments d'une variable structurée est retourné par la fonction $\text{len}()$.

Exemple : si $G = [5, 9, 12]$ alors $\text{len}(G) = 3$, $G[0] = 5$, $G[1] = 9$ et $G[2] = 12$.

Le tableau est de taille 3, mais l'indice le plus élevé est 2.

Pour répéter du code, par exemple pour parcourir les éléments d'un tableau, on peut utiliser une boucle **for**.

```
sum ← 0
n ← len(T)
for (i ← 0 to n - 1 step 1)
{
    sum ← sum + T[i]
}
```

La notation **for** ($i \leftarrow a$ **to** b **step** k) représente une boucle

- dans laquelle i commence à la valeur a ,
- qui sera répétée tant que $i \leq b$,
- qui augmente i de k à la fin de chaque étape.

L'exemple à gauche calcule la somme des éléments du tableau T .

Les éléments sont ajoutés un par un dans la variable sum .

Les instructions à exécuter dans une boucle seront clairement identifiées soit en les plaçant entre accolades $\{ \}$, soit en les décalant. Bien souvent, on fera les deux : mise entre accolades et décalage, comme ci-dessus.

On peut également écrire une boucle à l'aide de l'instruction **while** qui répète du code tant que sa condition est vraie.

Dans l'exemple à droite, on divise un nombre entier positif n par 2,

puis par 3, ensuite par 4 ...

jusqu'à ce qu'il ne soit plus composé que d'un seul chiffre

(c'est-à-dire jusqu'à ce que $n < 10$).

```
d ← 2
while (n ≥ 10) {
    n ← n/d
    d ← d+1
}
```

Question 1 – Le palais perdu épisode 1 : compter.

*Des archéologues ont découvert un ancien palais avec de nombreuses salles.
 Mais il y a des pièges qu'ils ne parviennent pas à éviter.
 Ils ont appelé le professeur Zarbi à la rescousse.*

Le professeur a vite compris qu'il fallait marcher sur certaines dalles gravées pour désactiver les pièges.
 Dans les salles, les dalles gravées sont toujours disposées en plusieurs rangées horizontales.
 La première rangée est constituée d'une seule dalle gravée, appelée le *sommet*.

Le professeur a découvert qu'il faut avancer en respectant certaines règles, sinon un piège se déclenche inévitablement.

1. Il faut démarrer du sommet (en haut sur les exemples) et marcher uniquement sur des dalles gravées.
2. En avançant, on doit toujours passer d'une dalle gravée à une autre qui la touche (pas seulement par un coin).
3. Il faut passer par une seule dalle gravée dans chaque rangée.
4. Il faut atteindre la dernière rangée de dalles gravées (tout en bas sur les exemples).

À partir de maintenant, le mot **chemin** signifie une traversée qui respecte ces 4 règles.
 À partir de maintenant, toutes les **dalles** citées ou dessinées sont des dalles gravées.

Les dalles sont souvent disposées en pyramide, comme dans tous les exemples sur cette page.
 On note **P_n** une pyramide constituée de n rangées de dalles.

Exemple 1

Voici des pyramides **P₃**, **P₄** et **P₅**.

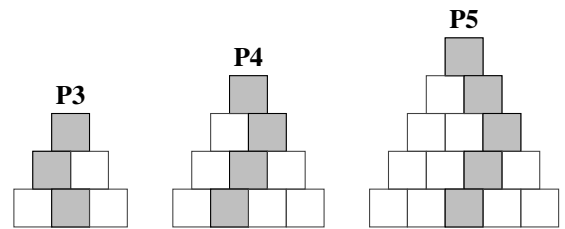
Un exemple de chemin est colorié en gris dans chacune.

Le sommet est en gris (règle 1).

Les dalles grises de rangées adjacentes se touchent (règle 2).

Il y a une seule dalle grise par rangée (règle 3).

Il y a une dalle grise dans la dernière rangée (règle 4).



Le professeur Zarbi veut compter combien de chemins différents permettent de traverser des pyramides.

Exemple 2

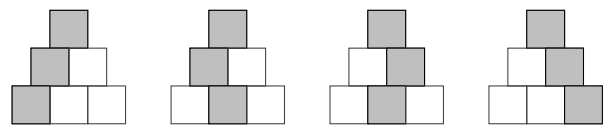
Voici tous les chemins qui traversent **P₃**.

1 chemin atteint la dalle de gauche de la dernière rangée.

2 chemins atteignent la dalle au centre de la dernière rangée.

1 chemin atteint la dalle de droite de la dernière rangée.

Au total, 4 chemins différents traversent **P₃**.



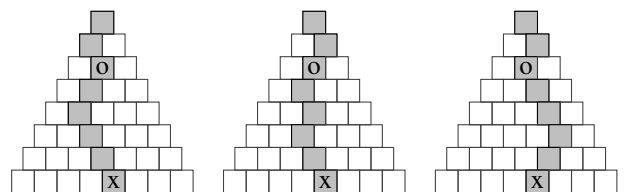
Parfois, il veut compter uniquement les chemins qui passent par certaines dalles.

Exemple 3

Voici 3 chemins qui traversent **P₈**.

Ils passent tous par la deuxième dalle de la troisième rangée (dalle marquée d'un o).

Ils atteignent tous la cinquième dalle de la dernière rangée (dalle marquée d'un x).

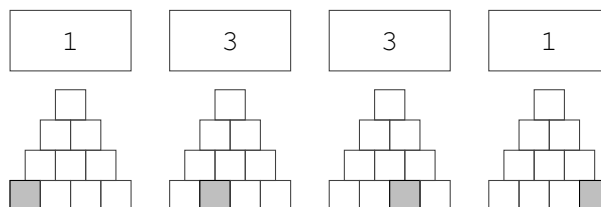


Remarque : sur une des dernières pages de ce questionnaire, des pyramides vides sont à votre disposition pour vos recherches au brouillon.

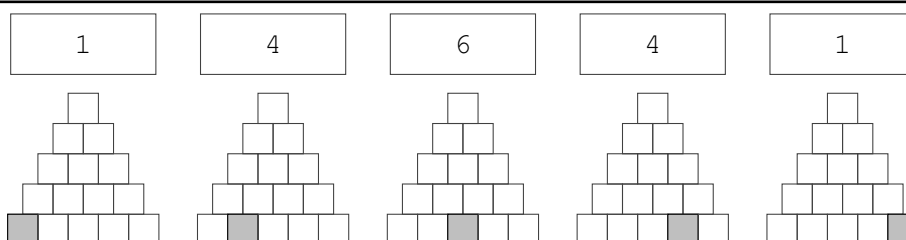
Compter les chemins qui atteignent une dalle précise.

Dans les questions suivantes, des pyramides sont dessinées avec 1 dalle grise dans la dernière rangée. Combien de chemins différents atteignent la dalle grise ? Écrivez les réponses dans les rectangles.

Q1(a) /4 Combien de chemins atteignent la dalle grise dans P4 ?



Q1(b) /5 Combien de chemins atteignent la dalle grise dans P5 ?

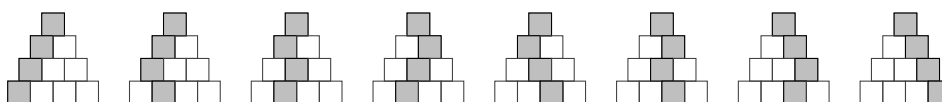


Compter tous les chemins.

Q1(c) /1 Au total, combien de chemins traversent P2 ? Solution : 2



Q1(d) /1 Au total, combien de chemins traversent P4 ? Solution : 8



Q1(e) /1 Au total, combien de chemins traversent P5 ? Solution : 16

Q1(f) /1 Au total, combien de chemins traversent P6 ? Solution : 32

Compter les chemins qui passent par certaines dalles.

Dans les questions suivantes, des pyramides sont dessinées avec une ou plusieurs dalles grises.

Écrivez dans le rectangle au-dessus de la pyramide le nombre de chemins différents qui passent par les dalles grises. N'oubliez pas les 4 règles !

Q1(g) /6 Combien de chemins passent par les dalles grises des pyramides P6 ?

$1 * 1 = 1$	$2 * 0 = 0$	$2 * 3 = 6$	$2 * 1 * 2 = 4$	$3 * 4 = 12$	$2 * 8 = 16$

Dalles disposées en diamant.

Les dalles sont souvent disposées en pyramide, mais ce n'est pas toujours le cas.

Il existe par exemple une disposition en diamant où la dernière rangée n'a qu'une seule dalle.

Dans les questions suivantes, des salles en diamant sont dessinées avec une ou plusieurs dalles grises.

Écrivez dans le rectangle au-dessus du diamant le nombre de chemins différents qui passent par les dalles grises.

Q1(h) /4 Combien de chemins passent par les dalles grises des diamants ?

1	9	8	20

Salles étranges.

Les dalles de certaines salles sont disposées de manière compliquée.

Dans les questions suivantes, il faut compter tous les chemins qui traversent des salles étranges.

Écrivez dans le rectangle au-dessus de la salle le nombre total de chemins qui la traversent (jusqu'à une dalle grise).

Q1(i) /4 Au total, combien de chemins traversent ces salles étranges ?

4	6	50	96

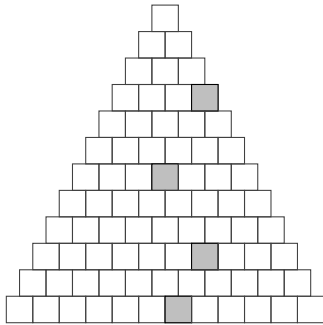
Chemins à étapes dans les grandes pyramides.

Dans les questions suivantes, de grandes pyramides sont dessinées avec plusieurs dalles grises.

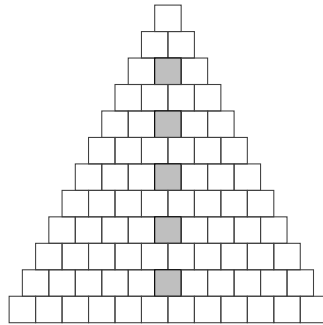
Écrivez dans le rectangle au-dessus de la pyramide le nombre de chemins différents qui passent par toutes les dalles grises. **N'oubliez pas les 4 règles !**

Q1(j) /6 Combien de chemins passent par les dalles grises des pyramides P12 ?

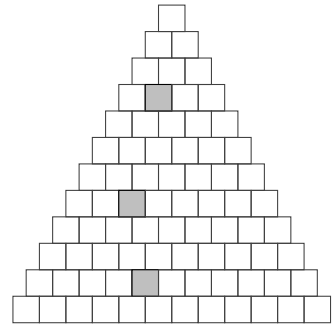
1



64

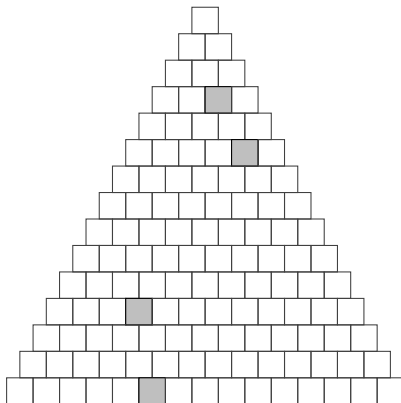


72

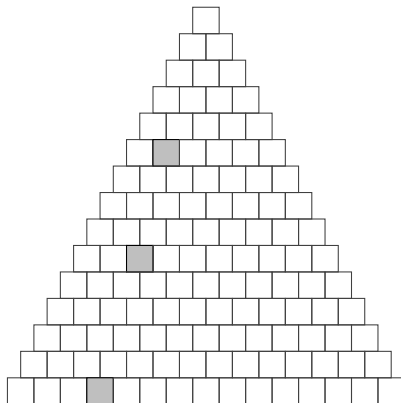


Q1(k) /6 Combien de chemins passent par les dalles grises des pyramides P15 ?

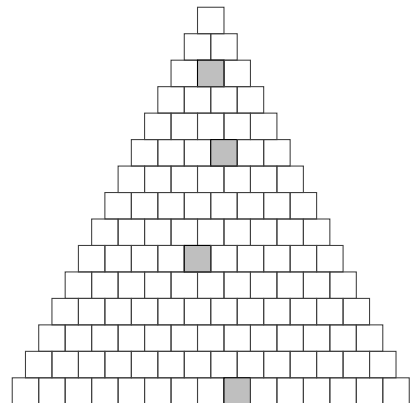
0



100

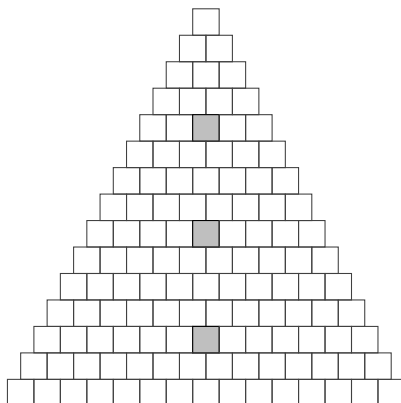


120



Q1(l) /4 Combien de chemins passent par les dalles grises de la pyramide P15 ?

864



Question 2 – Le palais perdu épisode 2 : décrypter.

*Les découvertes du professeur Zarbi ne permettent pas encore d'éviter tous les pièges.
 Mais les archéologues ont découvert les plans des salles parmi d'autres documents importants.
 Dans ces plans, une pyramide marquée de petits disques est dessinée sur chaque dalle.
 Le professeur comprend que ces pyramides représentent des nombres et il parvient à les décrypter.*

Attention ! Ne confondez pas !

Dans l'épisode 1, les pyramides étaient constituées de dalles.

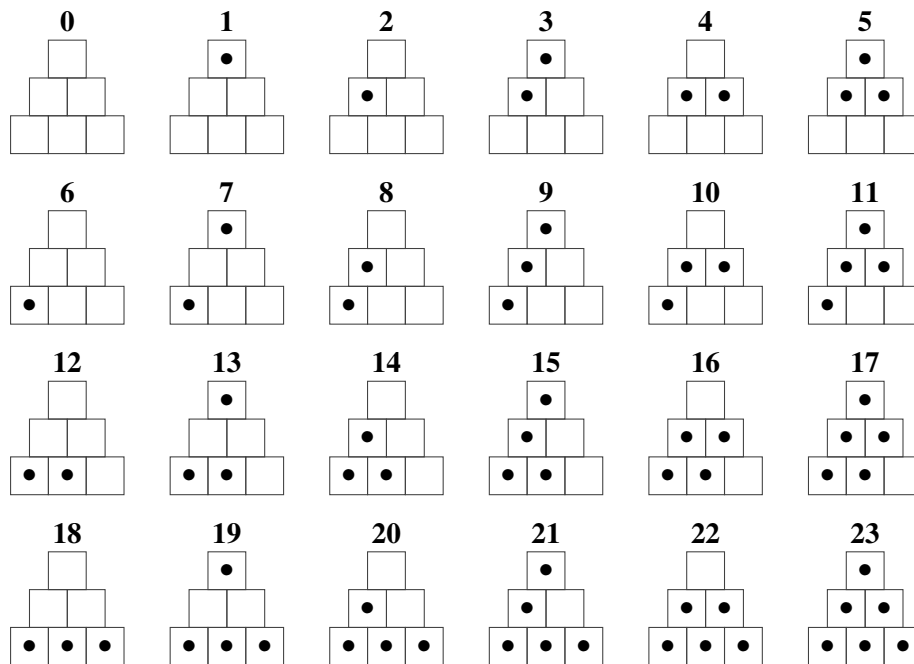
Dans l'épisode 2, les pyramides sont des nombres marqués sur les dalles.

Le système de numérotation.

Les nombres sont représentés par des pyramides dont certaines cases contiennent un petit disque noir.

- 0 est représenté par une pyramide sans aucun disque.
- Pour passer d'une pyramide à la suivante, c'est-à-dire d'un nombre au suivant, on procède comme suit.
 1. En partant du sommet, on cherche la première rangée qui contient une case vide.
 2. On place un petit disque dans n'importe quelle case vide de cette rangée.
 3. On vide toutes les cases des rangées supérieures (celles qui étaient remplies).

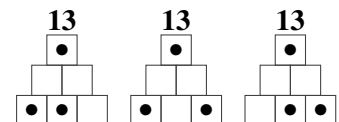
Exemples : voici des pyramides P3 qui représentent les nombres de 0 à 23.



Nombres qui ont plusieurs représentations.

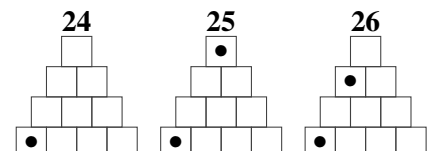
Plusieurs pyramides peuvent représenter le même nombre puisqu'on peut placer les disques noirs n'importe où dans leur rangée (point 2 ci-dessus).

Exemple : les 3 pyramides ci-contre représentent toutes le nombre 13.



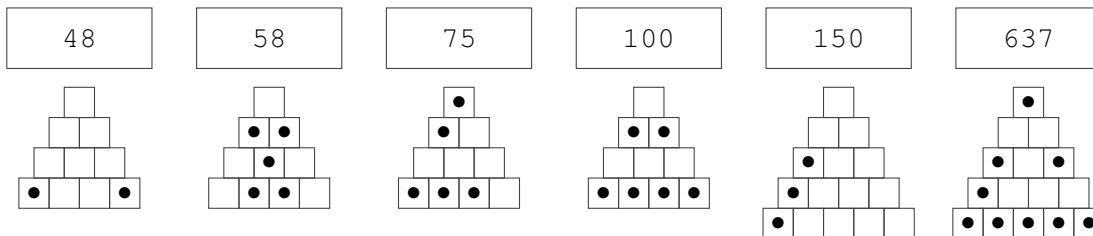
Représentations de nombres plus grands.

23 est le plus grand nombre qu'on peut représenter par une pyramide P3. Pour aller plus loin, il faut utiliser des pyramides avec plus de rangées.

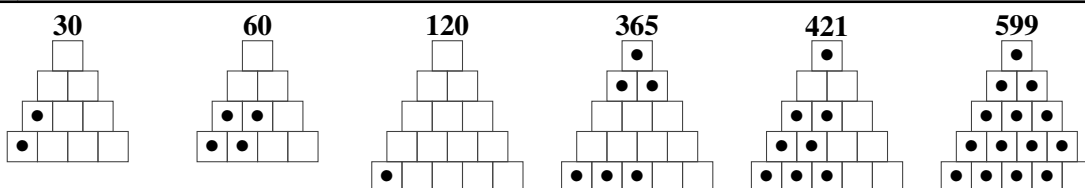


Remarque : *dessiner une pyramide* signifie *dessiner des petits disques noirs dans certaines cases de celle-ci*.
 Cas particulier : si on ne dessine aucun disque, on *dessine une pyramide vide* qui représente le nombre 0.

Q2(a) /6 Écrivez dans chaque rectangle le nombre représenté par la pyramide dessinée.



Q2(b) /6 Dessinez une pyramide qui représente le nombre indiqué.



Plus haut, on a fait remarquer que plusieurs pyramides peuvent représenter le même nombre et que 23 est le plus grand qu'on peut représenter par une pyramide P3. Cela inspire les questions suivantes auxquelles vous pouvez répondre par un nombre ou par une expression mathématique.

Q2(c) /1 Combien de pyramides P3 différentes peut-on dessiner ?
 Solution : $2^6 = 64$

Q2(d) /1 Combien de pyramides P4 différentes peut-on dessiner ?
 Solution : $2^{10} = 1024$

Q2(e) /1 Quel est le plus grand nombre qu'on peut représenter avec une pyramide P4 ?
 Solution : $5! - 1 = 119$

Q2(f) /1 Combien de pyramides P5 différentes peut-on dessiner ?
 Solution : $2^{15} = 32768$

Q2(g) /1 Quel est le plus grand nombre qu'on peut représenter avec une pyramide P5 ?
 Solution : $6! - 1 = 719$

Dans les questions suivantes, vous devez donner tous les nombres qui ont une certaine propriété.

S'il y a plusieurs nombres, écrivez-les en les séparant par des virgules.

S'il n'y a pas de nombre, répondez en traçant une croix X.

Q2(h) /1	Quels nombres ont une seule représentation dans P3 ? Solution : 0, 1, 4, 5, 18, 19, 22, 23
Q2(i) /1	Quels nombres ont exactement 2 représentations dans P3 ? Solution : 2, 3, 20, 21
Q2(j) /1	Quels nombres ont exactement 3 représentations dans P3 ? Solution : 6, 7, 10, 11, 12, 13, 16, 17
Q2(k) /1	Quels nombres ont exactement 4 représentations dans P3 ? Solution : X
Q2(l) /1	Quels nombres ont exactement 5 représentations dans P3 ? Solution : X
Q2(m) /1	Quels nombres ont exactement 6 représentations dans P3 ? Solution : 8, 9, 14, 15

Nouvelle notation : liste d'une pyramide.

Dessiner les pyramides est fastidieux et peu pratique.

Au lieu de dessiner une pyramide, le professeur Zarbi préfère donner la liste, en partant du sommet, des nombres de disques noirs dans chaque rangée.

À partir de maintenant, le **nombre d'une pyramide** est le nombre représenté par la pyramide. C'est un nombre entier positif, noté comme d'habitude avec les chiffres de 0 à 9.

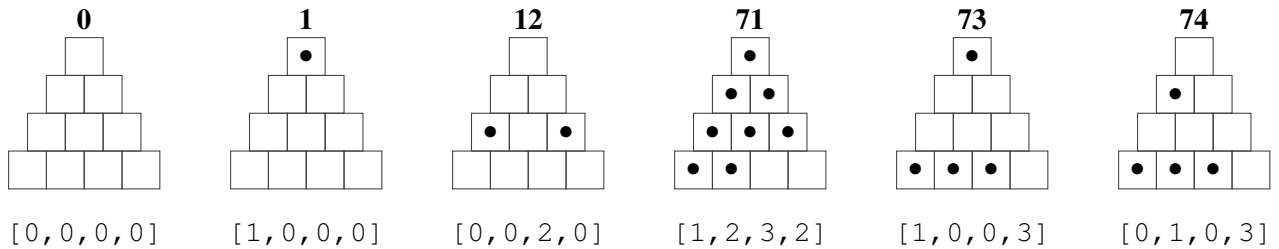
À partir de maintenant, la **liste d'une pyramide** est la liste des nombres de disques noirs dans chaque rangée de la pyramide.

Cette liste commence par 0 s'il n'y a pas de disque noir au sommet ou par 1 s'il y en a un.

Cette liste contient n nombres pour une pyramide **P_n**.

Exemples : voici quelques pyramides P₄.

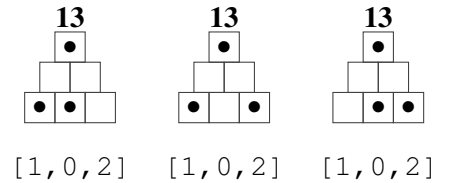
Pour chacune, son **nombre** est indiqué au-dessus et sa **liste** en-dessous.



Remarque.

Si plusieurs pyramides représentent le même nombre, alors elles ont la même liste.

Exemple : les 3 pyramides **P₃** qui représentent le nombre 13 ont la même liste [1, 0, 2].



Automatisation.

Le professeur doit souvent calculer le nombre d'une pyramide à partir de sa liste.

Il lui arrive aussi, inversement, de calculer la liste d'une pyramide à partir de son nombre.

Il veut écrire des programmes pour automatiser ses calculs.

Quelques rappels.

- **Division entière.**

L'opérateur / (ou // si vous préférez) donne le quotient entier d'une division entre 2 entiers.

L'opérateur % donne le reste d'une division entre 2 entiers.

Exemple : 14/3 est égal à 4 (de même que 14//3) et 14%3 est égal à 2.

- **Numérotation des éléments d'une liste.**

Les éléments d'une liste sont numérotés à partir de 0.

Exemple : si LIS=[1, 0, 2] alors LIS[0]=1, LIS[1]=0 et LIS[2]=2.

- **Initialisation d'une liste.**

LIS← [0] *n crée une liste de n zéros.

Fonction PyrLIS (n, dec)

Cette fonction reçoit 2 paramètres.

- n : le nombre de rangées de la pyramide.
- dec : un nombre entier positif qu'il faut représenter par une pyramide.
 dec est nommé ainsi pour rappeler qu'il s'agit d'un nombre **décimal** (c'est-à-dire écrit avec les 10 chiffres de 0 à 9).

La fonction retourne la liste d'une pyramide **P_n** qui représente le nombre dec .

Exemples : `PyrLIS(3, 13)` retourne `[1, 0, 2]` et `PyrLIS(4, 13)` retourne `[1, 0, 2, 0]`.

La question suivante rapporte entre 0 et 6 points.

Vous perdez 3 points pour chaque zone qui n'est pas correctement complétée.

Q2(n) /6 Complétez les dans la fonction `PyrLIS(dec, n)`.

```
function PyrLIS(n, dec) {
  LIS ← [0]*n
  for r ← 0 to n-1 step 1 {
    LIS[r] ← 
  }
  dec ← 
}
return(LIS)
}
```

Fonction Pyrdec (n, LIS)

Cette fonction reçoit 2 paramètres.

- n : le nombre de rangées de la pyramide.
- LIS : la liste d'une pyramide.

La fonction retourne le nombre dec de la pyramide **P_n** dont la liste est donnée.

Exemples : `Pyrdec(3, [1, 0, 2])` et `Pyrdec(4, [1, 0, 2, 0])` retournent 13.

La question suivante rapporte entre 0 et 6 points.

Vous perdez 3 points pour chaque zone qui n'est pas correctement complétée.

Q2(o) /6 Complétez les dans la fonction `Pyrdec(LIS, n)`.

```
function Pyrdec(n, LIS) {
  dec ← 0
  v ← 1
  for r ← 0 to n-1 step 1 {
    v ← v * 
  }
  dec ← 
}
return(dec)
}
```

Question 3 – Le palais perdu épisode 3 : additionner.

Le professeur Zarbi a décrypté tous les plans où de petites pyramides à disques étaient dessinées sur les dalles. Il a recopié les plans en remplaçant chaque petite pyramide par le nombre qu'elle représente. Après de longues réflexions et des expérimentations minutieuses, il a enfin trouvé la solution...

Attention ! Ne confondez pas !

Dans l'épisode 3, les pyramides sont à nouveau constituées de dalles, comme dans l'épisode 1. Mais dans cet épisode, chaque dalle porte un nombre (celui décrypté dans l'épisode 2).

Chemins de somme maximale.

À partir de maintenant, une *pyramide de nombres* est une pyramide avec un nombre sur chaque dalle.

Pour rappel, un *chemin*

- démarre du sommet,
- passe d'une dalle à une autre qui la touche,
- passe sur une seule dalle dans chaque rangée,
- doit atteindre la dernière rangée.

Le *score* d'un chemin est la somme des nombres marqués sur les dalles de ce chemin.

Le professeur Zarbi a découvert que le chemin qui a le plus grand score ne déclenche pas de piège.

Si plusieurs chemins ont le même score maximal, aucun ne déclenche de piège.

Tout chemin qui a un score inférieur au score maximal déclenche un piège.

Exemples : voici quelques pyramides **P5**.
 Pour chacune, un chemin de score maximal est colorié en gris.
 Le score du chemin est indiqué dans le rectangle.

$7+6+3+4+9=29$	$10+16+8+19+18=71$	$7+5+11+17+15=55$	$1+1+5+1+17=25$
----------------	--------------------	-------------------	-----------------

Dans la question suivante, des pyramides sont dessinées avec un nombre sur chaque dalle.

Dans chacune, tracez un chemin de score maximal (en entourant les nombres ou en coloriant les dalles au crayon).

Écrivez le score du chemin tracé dans le rectangle au-dessus de la pyramide.

Q3(a) /8	Tracez un chemin de score maximal et indiquez son score dans le rectangle.		
-----------------	---	--	--

64	35	40	44
----	----	----	----

Pyramides à utiliser pour vos recherches au brouillon.

