

Association France-ioi

L'apprentissage ludique de la programmation, chacun à son rythme

info @ france-ioi.org



Qui sommes-nous ?

- Association loi 1901
- Créée en 2004
- 30 bénévoles passionnés d'algorithmique et d'enseignement

- Promouvoir l'apprentissage de la programmation et de l'algorithmique
- Concevoir des outils pour apprendre à son rythme par la pratique

Les outils



CASTOR

castor-informatique.fr



lgoréa

algorea.org



France-IOI

france-ioi.org



ALKINDI

concours-alkindi.fr



parcours.algorea.org

CH ' T I CODE

chticode.algorea.org

code : chticode



CASTOR INFORMATIQUE FRANCE

Annonce : le concours Castor 2017 est terminé. Les [résultats sont disponibles](#)

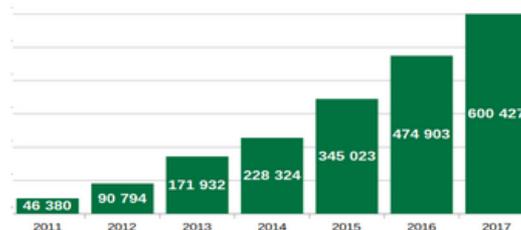
Le 1er tour du concours de programmation [Algoréa](#) est en cours.

Le concours Castor vise à faire découvrir aux jeunes l'informatique et les sciences du numérique.

Le concours est organisé tous les ans, au mois de novembre. Il se déroule sous la supervision d'un enseignant, en salle informatique.

[S'entraîner sur les sujets passés](#)

Nombre de participants au concours Castor



RÉSULTATS 2017

L'édition 2017 a été un grand succès, avec plus de 600 000 participants dans 3284 établissements !

Les participants peuvent [accéder à leur score](#) en utilisant leur code d'équipe fourni au début du concours, ou leur code de participant.

Les coordinateurs peuvent accéder aux scores et classements de tous leurs élèves sur [l'interface coordinateur](#).

MENU

[Présentation](#)

[Accès aux sujets](#)

[Accès aux résultats](#)

[Accès coordinateur](#)

[Manuel du concours](#)

[Qui sommes nous ?](#)

[Contact](#)

[Informations légales](#)

RÉSEAUX

Rejoignez-nous sur :



Facebook

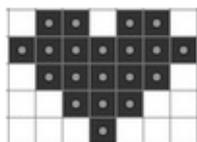


Twitter

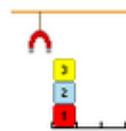
Langage de formes



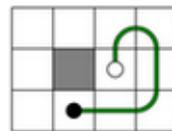
Copie rapide



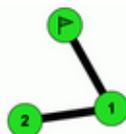
Tour infernale



Chemins distincts



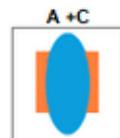
Accrobranche



Fantômes



Formes combinées



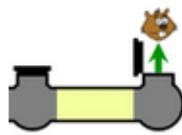
Immeuble éclairé



Photos du jeu



Tas de graines



Constellation



Poésie

Excellent nageur

Et bon constructeur

Grâce à mes dents

Merveilleux instruments



Trois versions pour chaque question



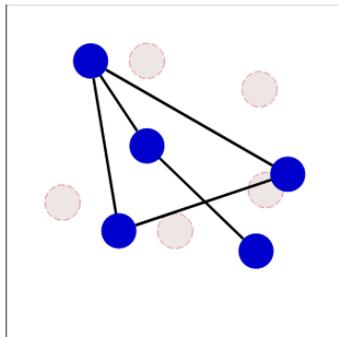
Constellation ☆☆☆☆

Version ☆☆☆

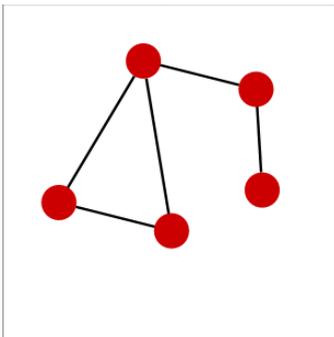
Version ☆☆☆☆

Version ☆☆☆☆☆

Déplacez les ronds bleus de gauche pour obtenir exactement la figure de droite.



Objectif



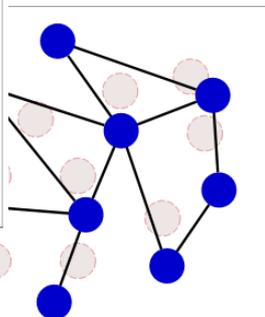
Constellation ☆☆☆☆

Version ☆☆☆

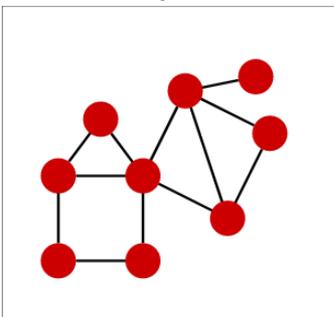
Version ☆☆☆☆

Version ☆☆☆☆☆

Déplacez les ronds bleus de gauche pour obtenir exactement la figure de droite.



Objectif



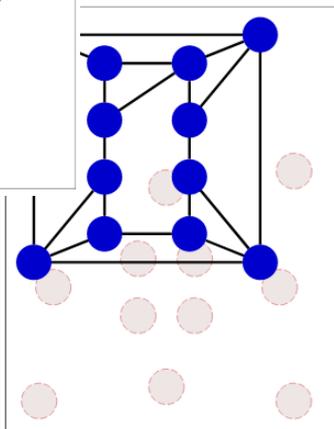
Constellation ☆☆☆☆

Version ☆☆☆

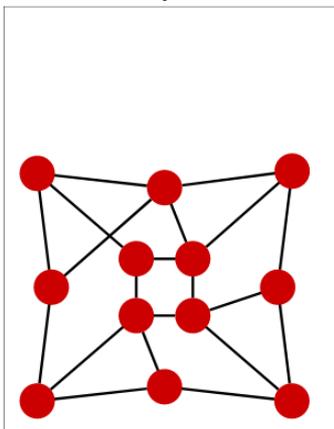
Version ☆☆☆☆

Version ☆☆☆☆☆

Déplacez les ronds bleus de gauche pour obtenir exactement la figure de droite.



Objectif



C'est de l'informatique !

Ce défi illustre le problème de faire correspondre un **graphe**, avec un autre graphe ayant la même structure.

Ce problème est connu sous le nom d'**isomorphisme de graphe**.



CASTOR INFORMATIQUE FRANCE

PLATEFORME DU CONCOURS - **ACCÈS COORDINATEUR**

Se déconnecter

Explications

Coordinateur

Établissement

Groupes

Équipes

Élèves

Récompenses

Diplômes

Création des groupes

Le système mis en place repose sur la création de groupes d'élèves. Cela réduit le nombre d'informations que vous donnez avant le concours (vous déclarez des groupes et non pas tous les élèves). Cela réduit aussi le nombre de codes d'accès au concours (un par groupe).

Ainsi, afin de préparer le passage du concours dans votre établissement :

1. Établissez la liste des « groupes de passage » dont votre établissement a besoin.

Un « groupe » réunit des élèves d'un niveau sur une tranche horaire. Ainsi, par exemple, si vous faites passer tous les élèves de la 6^e2 ensemble à un moment, c'est un groupe. Si vous faites passer une première moitié de la 6^e2 à un moment, et l'autre moitié à un autre moment, cela fait deux groupes. Des élèves de 6^e différentes passant à la même heure, c'est un seul groupe. Si vous faites passer des élèves de 6^e et des élèves de 4^e en même temps, ils constituent un deuxième groupe.

2. Dans l'onglet GROUPE, créez les groupes : déclarez le nom, le niveau de classe, la date à laquelle le groupe passera le concours, et la taille de chaque groupe.

Concours Algoréa

Présentation

Participer

Se préparer

Récompenses

Résultats

Les classements du 1er tour du concours Algoréa 2018 [sont disponibles](#) jusqu'à la catégorie orange.

La catégorie verte du 1er tour est ouverte jusqu'au 28 février, pour les qualifiés. Une 2e [préparation](#) est aussi disponible.

Les trois premiers tours se déroulent sur le [même site que le Castor](#)

Le concours Algoréa, c'est quoi ?

Pour encourager les jeunes français et françaises à progresser en programmation et algorithmique, [France-ioi](#) organise un grand concours ouvert à tous : le **concours Algoréa**. Plus de 210 000 élèves participent à l'édition 2018.

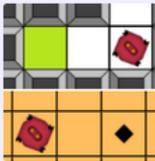
Une progression en couleurs

Blanc	Séquence d'instructions, appels de fonctions, boucles "répéter n fois".
Jaune	Conditions si ... alors et si ... alors ... sinon, imbrications de boucles.
Orange	Variables simples, boucles tant que, opérateurs arithmétiques (+, -, *, /), opérateurs booléens (ET, OU, NON)
Vert	Déclarations de fonctions, tableaux à 1 dimension ou listes, chaînes de caractères, modulo, nombres à virgule.
Bleu	Bases de la complexité algorithmique, tableaux à plusieurs dimensions, types composés, structures de données élémentaires (pile, file), balayages élémentaires, récursivité, tris simples.
Marron	Manipulation d'arbres, structures de données et balayages, manipulation de graphes, graphes implicites, algorithmes semi-numériques.
Noir	Algorithmes gloutons, diviser pour régner, arbres binaires, tris efficaces, plus courts chemins, union-find, algorithmes dynamiques.
Rouge	Graphes implicites avancés, algorithmes dynamiques avancés, structure de données et balayages avancés, composantes fortement connexes, flots et couplages

Concours Algoréa

[Présentation](#)[Participer](#)[Se préparer](#)[Récompenses](#)[Résultats](#)

Blockly →	
Scratch →	
Python →	<pre>droite() droite() for loop in range (haut () gauche() if (obstacleDroi bas())</pre>

Préparation Algoréa Blanche Blockly, Peinture et Parcours →	
Préparation Algoréa Blanche Blockly, Pierres et Plots →	
Préparation Algoréa Blanche Blockly, Espace →	
Préparation Algoréa Blanche Blockly 2017 →	

Ramasser les objets dans l'espace 2 ☆☆☆☆

Version ☆☆

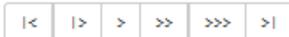
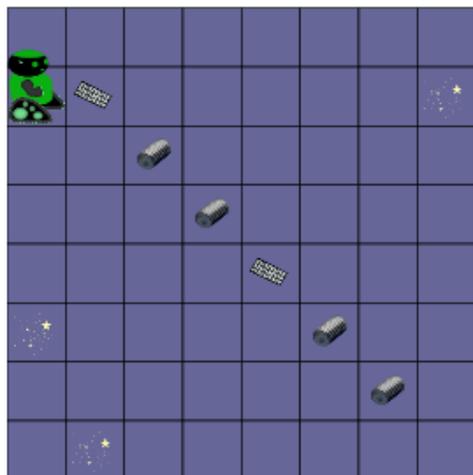
Version ☆☆☆

Version ☆☆☆☆

Programmez le robot pour qu'il ramasse tous les objets perdus dans l'espace.

Vous pouvez avoir besoin du [bloc répéter](#)

4 blocs restants sur 10 autorisés.



Valider le programme

tourner à gauche

tourner à droite

avancer

ramasser l'objet

répéter 10 fois

faire

Programme du robot

répéter 8 fois

faire

avancer

ramasser l'objet

tourner à droite

avancer

tourner à gauche

Ramasser les objets dans l'espace 2 ☆☆☆☆

Version ☆☆

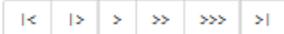
Version ☆☆☆

Version ☆☆☆☆

Programmez le robot pour qu'il ramasse tous les objets perdus dans l'espace.

Vous pouvez avoir besoin du [bloc répéter](#)

4 blocs restants sur 10 autorisés.



Valider le programme

tourner à gauche

tourner à droite

avancer

ramasser l'objet

répéter 10 fois

Quand cliqué

répéter 7 fois

avancer

ramasser l'objet

tourner à droite

avancer

tourner à gauche

Ramasser les objets dans l'espace 2 ☆☆☆☆

Version ☆☆

Version ☆☆☆

Version ☆☆☆☆

Programmez le robot pour qu'il ramasse tous les objets perdus dans l'espace.

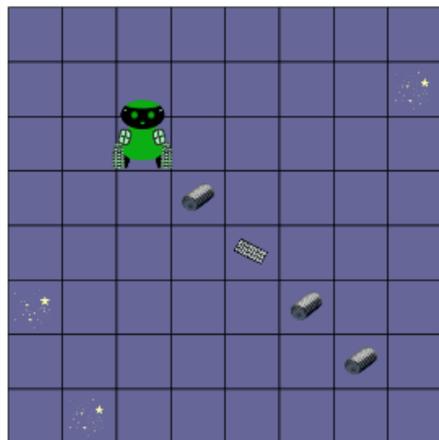
Vous pouvez avoir besoin de [la boucle for](#)

Fonctions disponibles : `tournerGauche()`, `tournerDroite()`, `avancer()`, `ramasserObjet()`.

Mots-clés autorisés : `for`.

[Plus de détails](#)

4 blocs restants sur 10 autorisés.



Valider le programme

```
1 from robot import *
2
3 for loop in range(6):
4     avancer()
5     ramasserObjet()
6     tournerDroite()
7     avancer()
8     tournerGauche()
```

Sujet catégorie orange version 2*

Version ☆☆☆

Version ☆☆☆

Version ☆☆☆

Programmez le robot pour qu'il peigne toute les cases marquées avec un point noir.

Le nombre devant le robot indique combien de cases sont marquées, un peu plus loin. Il doit donc lire ce nombre, le stocker dans sa mémoire grâce à une *variable*, puis utiliser le contenu de cette variable dans le programme.

Pour lire le nombre et stocker sa valeur dans la variable *nombreMarquées*, avancez sur sa case, puis exécutez l'instruction suivante :

fixer *nombreMarquées* à nombre sur la case

Vous pouvez ensuite utiliser cette variable là où vous avez besoin de sa valeur, par exemple :

répéter *nombreMarquées* fois
faire

Notez que cet exercice contient deux tests. Le programme du robot doit fonctionner sur chacun des deux tests.

Test 1

|< |> > >> >>> >|

Test 2

Valider le programme

15 blocs restants sur 15 autorisés.

Programme du robot

- avancer vers la droite
- peindre la case
- nombre sur la case
- répéter 10 fois
faire
- mémoire du robot
- fixer mémoire du robot à
- incrémenter mémoire du robot de 1

Le même sujet version 3*

Version ★★

Version ☆☆☆

Version ★★★★★

Programmez le robot pour qu'il peigne toute les cases marquées avec un point noir.

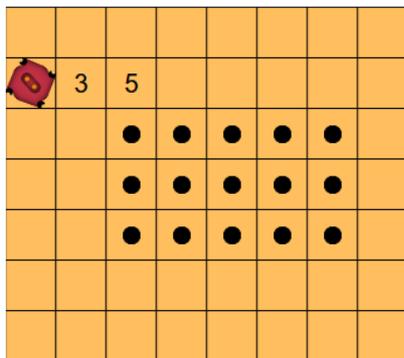
Devant le robot se trouvent deux nombres : le nombre de lignes et le nombre de colonnes du rectangle de cases marquées.

Pour résoudre cet exercice, vous devrez créer deux **variables**, pour stocker ces deux nombres. Appelez-les par exemple *nbLignes* et *nbColonnes*.

Dans cette version, les blocs sont regroupés par catégorie dans des menus. Cliquez sur un menu pour accéder aux blocs de la catégorie correspondante.

Notez que cet exercice contient deux tests. Le programme du robot doit fonctionner sur chacun des deux tests.

Test 1



|< |> > >> >>> >|

Test 2

Valider le programme

25 blocs restants sur 25 autorisés.

- Actions
- Capteurs
- Variables
- Boucles

Programme du robot



Version ★★

Version ★★★

Version ★★★★★

Programmez le robot pour qu'il peigne toute les cases marquées avec un point noir.

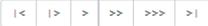
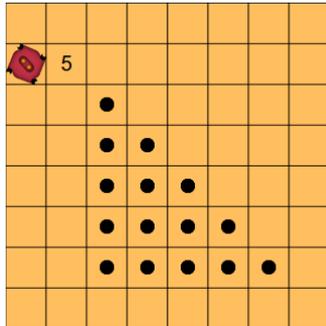
Devant le robot se trouve un nombre : le nombre de lignes et de colonnes occupées par le triangle de cases marquées.

Pour peindre le bon nombre de cases sur chaque ligne, vous allez devoir créer une **variable** qui pour chaque ligne, devra contenir le nombre de colonnes à marquer. Après avoir dessiné chaque ligne, le contenu de cette variable devra être augmenté de 1. Cela ressemblera un peu à ceci :

```
Programme du robot
fixer nbColonnes à 1
répéter 10 fois
faire
  avancer vers le bas
  incréments nbColonnes de 1
```

Notez que cet exercice contient deux tests. Le programme du robot doit fonctionner sur chacun des deux tests.

Test 1



Test 2

Valider le programme

30 blocs restants sur 30 autorisés.

- Actions
- Capteurs
- Variables
- Boucles
- Maths

Programme du robot

Connexion

Se connecter

Langue : 

Plan du site

Progresser

- Présentation
- Cours et problèmes
- Questions fréquentes
- Forum d'entraide

Enseigner

- Présentation
- Groupes et classes

Concourir

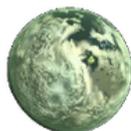
- Présentation
- Classement
- Épreuves de concours
- Résultats

Olympiades

- Présentation
- Sélection
- Résultats

Le site d'entraînement à la programmation et l'algorithmique

Notre objectif est de faire découvrir la programmation et l'algorithmique au plus grand nombre de personnes possible. Nous créons et diffusons gratuitement des outils et contenus permettant de progresser rapidement dans ces domaines, et organisons des concours pour accompagner cette progression.



Concours de programmation Algoréa

Concours en 5 tours de janvier à août pour progresser en programmation puis en algorithmique, ouvert aux collégiens et lycéens français. Plus de 47 000 élèves ont participé en 2017 !

Le 1er tour du concours Algoréa 2018 est en cours !

Apprentissage de la programmation et de l'algorithmique



Que vous soyez novice ou déjà expérimenté(e) en programmation, notre plateforme d'apprentissage vous permettra de bien maîtriser les aspects fondamentaux de la programmation, puis d'explorer à votre rythme le monde fascinant de l'algorithmique.

Commencez dès maintenant à progresser !



Concours Castor Informatique

Le concours Castor a pour but de faire découvrir les sciences du numérique, du CM1 à la terminale. Il est organisé par France-ioi, Inria et ENS Paris Saclay.

Plus de 600 000 élèves d'établissements français ont participé à l'édition 2017.

265 personnes connectées

Problèmes résolus

emmaboufree	15 h 55
Mathématiques de base	
emeline14 ²⁰²⁰	15 h 55
Bonbons pour tout le mo...	
marinet57	15 h 55
Transport d'eau	
loicmalo	15 h 55
Page d'écriture	
boontease	15 h 55
Dans le fourré	
vlacas	15 h 54
Mathématiques de base	
lanona02	15 h 54
Plan de la montagne	
emeric75	15 h 53
Banquet municipal	
aaaaaaaantoine	15 h 53
Jeu de dames	
emeline14 ²⁰²⁰	15 h 53
Vendanges	

Cours et problèmes

Voir les cours et résoudre les problèmes en :

C	C++	Pascal	OCaml	Java	JavaScool	Python
---	-----	--------	-------	------	-----------	--------

Parcours général	Parcours lycée	Méthodes	Problèmes non classés
------------------	----------------	----------	-----------------------

Le parcours général permet de progresser à son rythme, en partant des bases de la programmation et en allant jusqu'à un niveau avancé en algorithmique.

Niveau 1

‣ 1 - Affichage de texte, suite d'instructions	6 problèmes
‣ 2 - Répétitions d'instructions	10 problèmes
‣ 3 - Calculs et découverte des variables	13 problèmes
‣ 4 - Lecture de l'entrée	10 problèmes
‣ 5 - Tests et conditions	8 problèmes
‣ 6 - Structures avancées	8 problèmes
‣ 7 - Conditions avancées, opérateurs booléens	10 problèmes
‣ 8 - Répétitions conditionnées	5 problèmes



Python



Invasion de batraciens

Entraînement



Sujet

Résoudre

Conseils

Activité

Correction

L'automne ayant été très pluvieux, les Bufo Algo, une espèce locale de crapauds, se sont reproduits en grand nombre, et les habitants ont constaté que leur nombre doublait chaque semaine ! Leurs prédateurs naturels, les couleuvres (un type de serpent), sont complètement dépassés !

Avoir trop de crapauds est très gênant (on ne peut plus dormir) ! Les villageois décident donc d'élever un grand nombre de couleuvres avec lesquelles ils pourront contrôler le nombre de crapauds. Il leur faut tout d'abord estimer le nombre de crapauds qu'il y aura au cours des semaines qui viennent. Ils vous demandent votre aide.

Ce que doit faire votre programme :

Sachant qu'il y a actuellement 1337 crapauds et que leur nombre double chaque semaine, votre programme devra afficher le nombre de crapauds qu'il y aura après la 12^e semaine.

Important : vous devez **utiliser une boucle** pour calculer le nombre de crapauds.

 Python  Invasion de batraciens Entraînement  

Sujet Résoudre Conseils Activité Correction

Afficher/cacher les programmes minimaux à compléter

Éditer son programme

```
1 nombreCrapauds=1337
2 for loop in range (12):
3     nombreCrapauds=2*nombreCrapauds
4 print (nombreCrapauds)
```



Python



Invasion de batraciens

Entraînement



Sujet

Résoudre

Conseils

Activité

Correction

Conseils automatiques

0 conseil demandé sur 2 disponibles :

- On souhaite obtenir le nombre de crapauds au bout d'un certain nombre de semaines, sachant qu'il double chaque semaine. Il faut donc une structure de ce type :

```
Répéter pour chaque semaine
  Doubler le nombre de crapauds
Afficher le nombre de crapauds .
```

- On représente le nombre de crapauds par une variable, mettons *nbCrapauds*. Pour que sa valeur double dans la boucle, il faut lui en affecter une nouvelle :

```
nbCrapauds <- ...
```

de sorte que la variable contienne successivement les valeurs 1 337, 2 674, 5 348, 10 696...

 Python  **Invasion de batraciens** Entraînement  

Sujet Résoudre Conseils **Activité** Correction

Total des points obtenus sur ce sujet : 100.

Sujet commencé le 10/06/2013 à 18 h 55.

- ▶  (100) Soumission n° 1 le 10/06/2013 à 18 h 59 en C++
- ▶  (100) Soumission n° 2 le 02/02/2016 à 11 h 26 en Python



Python



Invasion de batraciens

Entraînement



Sujet

Résoudre

Conseils

Activité

Correction

Histoire

Presque 5,5 millions de crapauds, c'est énorme ! Heureusement, les couleuvres vont y remédier rapidement.

Algorithme

On utilise une variable nommée *nbCrapauds* pour stocker le nombre actuel de crapauds. Au début, ce nombre est égal à 1337. Pour faire doubler ce nombre à chaque semaine, on utilise une affectation pour mettre la valeur $nbCrapauds \times 2$ dans la variable *nbCrapauds*. On place cette affectation dans une boucle pour la répéter 12 fois de suite. On obtient alors le nombre total de crapauds à la fin des 12 semaines.

Programme

Pseudo-code

C

C++

Pascal

OCaml

Java

JavaScool

Python

```
nbCrapauds <- 1337
Répéter 12 fois
  nbCrapauds <- nbCrapauds * 2
Afficher nbCrapauds
```



Groupe : C++Mania

C++Mania, groupe de type « Classe » créé le 26/10/2014.

Membres du groupe

- [redacted] (en attente de validation)
- [redacted] — Superviseur
- [redacted] — Administrateur
- [redacted] — Superviseur

Progression des membres

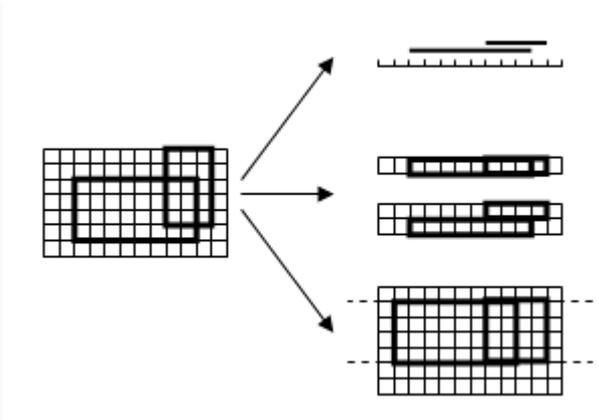
Cliquez sur une ligne pour obtenir des détails.

Répétitions conditionnées

	[redacted]	[redacted]
Département de médeci... Découverte	10 essais — 31 min	17 essais — 34 j 22 h 59 min
Administration : comp... Entraînement	2 essais — 13 min	3 essais — 4 min
Département de péda... Entraînement	2 essais — 14 min	1 essai — 6 min
Département d'architect... Validation	4 essais — 15 min	1 essai — 15 min
Département de chimie : ... Validation	9 essais — 24 min	3 essais — 9 min



1. Lisez bien le sujet, et reformulez-le.
2. Faites la liste des dimensions du sujet.
3. Cherchez une bonne représentation visuelle du problème.
4. Générez des exemples, et résolvez-les entièrement à la main.
5. Décrivez la solution naïve, puis essayez de l'améliorer.
6. Simplifiez le problème, puis généralisez les solutions obtenues.
7. Changez de point de vue, en envisageant les algorithmes classiques.



Tutorat, stage Toussaint 2017

Exemple d'application : population sur un rectangle

Dans une grille de nombres, répondre à de nombreuses requêtes de type :

“quelle est la population totale au sein de ce rectangle ?”

21	5	37	46	12
14	6	7	2	31
27	28	44	9	45
31	11	18	28	32
14	1	25	13	23

Exemple de requête

21	5	37	46	12
14	6	7	2	31
27	28	44	9	45
31	11	18	28	32
14	1	25	13	23

Pour chaque requete (xMin, xMax, yMin, yMax) :

Somme = 0

Pour x allant de xMin à xMax

 Pour y allant de yMin à yMax

 Somme = Somme + valeur[x][y]

Afficher Somme

Objectif : répondre à de nombreuses requêtes de type :
“quelle est la population totale au sein de ce segment

21	5	37	46	25	12	6	17	30
----	---	----	----	----	----	---	----	----

Solution : tableau cumulatif

21	5	37	46	25	12	6	17	30
----	---	----	----	----	----	---	----	----

0	21	$21+5$ 26	$26+37$ 63	$63+46$ 109	$109+25$ 134	$134+12$ 146	$146+6$ 152	$152+17$ 169	$169+30$ 199
---	----	--------------	---------------	----------------	-----------------	-----------------	----------------	-----------------	-----------------

Population sur un rectangle

21	5	37	46	12
14	6	7	2	31
27	28	44	9	45
31	11	18	28	32
14	1	25	13	23

Somme sur les lignes :

21	$21+5$ 26	$26+37$ 63	$63+46$ 109	$109+12$ 121
14	$14+6$ 20	$20+7$ 27	$27+2$ 29	$29+31$ 60
27	$27+28$ 55	$55+44$ 99	$99+9$ 108	$108+45$ 153
31	$31+11$ 42	$42+18$ 60	$60+28$ 88	$88+32$ 120
14	$14+1$ 15	$15+25$ 40	$40+13$ 53	$53+23$ 76

Population sur un rectangle

21	5	37	46	12
14	6	7	2	31
27	28	44	9	45
31	11	18	28	32
14	1	25	13	23

Sommes sur les lignes :

21	26	63	109	121
14	20	27	29	60
27	55	99	108	153
31	42	60	88	120
14	15	40	53	76

0	0	0	0	0	0
0	21	26	63	109	121
0	21+14 35	20+26 46	27+63 90	109+29 138	121+60 181
0	35+27 62	55+46 101	90+99 189	108+138 246	153+181 334
0	62+31 93	101+42 143	189+60 249	246+88 334	334+120 454
0	93+14 107	15+143 158	249+40 289	334+53 387	454+76 530

Population sur un rectangle

21	5	37	46	12
14	6	7	2	31
27	28	44	9	45
31	11	18	28	32
14	1	25	13	23

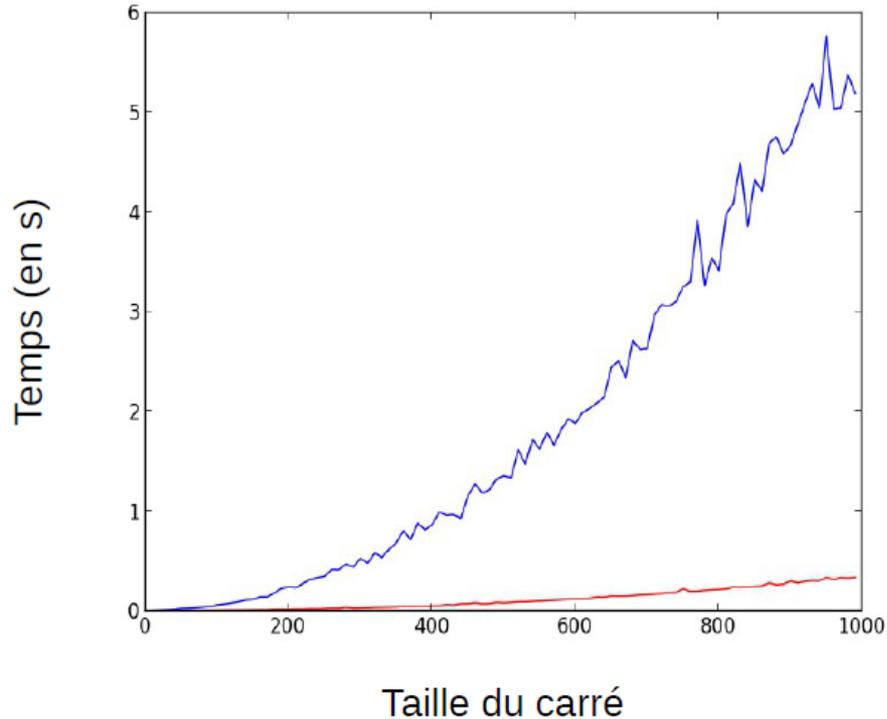
0	0	0	0	0	0
0	21	26	63	109	121
0	35	46	90	138	181
0	62	101	189	246	334
0	93	143	249	334	454
0	107	158	289	387	530

La somme sur une requête est :
sommeAvant[linMax][colMax]
-sommeAvant[linMin-1][colMax]
-sommeAvant[linMax][colMin-1]
+sommeAvant[linMin-1][colMin-1]

Population sur un rectangle : performances



Nombre de requêtes : 1000
Les requêtes sont choisies aléatoirement



Rouge :
 $\text{nbRequetes} * \text{nbCases}$

Bleu :
 $\text{nbRequetes} + \text{nbCases}$



CONCOURS



ALKINDI

Accueil

Manuel du concours

Les épreuves précédentes

Découvrir la crypto

Enseignants

Qui sommes-nous ?

Le premier tour de l'édition 2017-2018 est terminé. Les coordinateurs ont accès aux résultats..

Les seuils de qualification au 2e tour sont disponibles dans le [manuel du concours](#)

Le deuxième tour est ouvert jusqu'au 10 mars. Il se déroule sur suite.concours-alkindi.fr





Parcours > Parcours officiels > Parcours Algorea > Chapitre 3 > Invasion de batraciens

Invasion de batraciens

ÉNONCÉ & RÉOLUTION

S'entrainer – utiliser une variable dans une boucle



Il y a actuellement 1337 crapauds et leur nombre double chaque semaine.

Ton programme doit calculer et afficher le nombre de crapauds qu'il y aura dans 12 semaines.

Code1 x + ⌂

- Variables
- Texte
- Maths
- Boucles

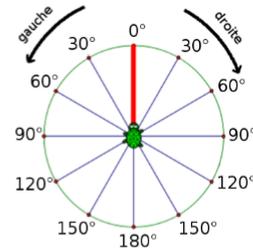
Programme du robot

Langage du fichier : Blockly



Boucles imbriquées 4.2 ☆☆☆☆☆

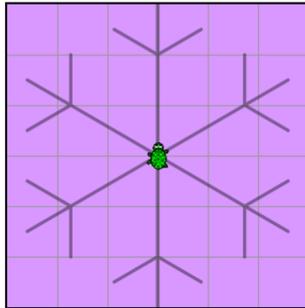
Votre tortue doit se déplacer sur le trait gris et y peindre sa trace.



Fonctions disponibles : `avancer()`, `reculer()`, `gauche()`, `droite()`, `couleur()`.

Mots-clés autorisés : `for`.

[Plus de détails](#)



Valider le programme

25 blocs restants sur 25 autorisés.

```
1 from turtle import *  
2
```