

# Que faire en algorithmique en classe de seconde ?

EIHassan FADILI  
Lycée Salvador Allende

# BEGIN

EiHassan FADILI  
Lycée Salvador Allende

# Que dit le programme ?

EiHassan FADILI  
Lycée Salvador Allende

# Algorithmique (objectifs pour le lycée)

- La démarche algorithmique est, depuis les origines, une composante essentielle de l'activité mathématique.
- Au collège, les élèves ont rencontré des algorithmes (**algorithmes opératoires, algorithme des différences, algorithme d'Euclide, algorithmes de construction en géométrie**).

# Algorithmique (objectifs pour le lycée)

- Ce qui est proposé dans le programme est une **formalisation en langage naturel** propre à donner lieu à **traduction sur une calculatrice ou à l'aide d'un logiciel.**
  
- Il s'agit de familiariser les élèves avec les grands principes d'organisation d'un algorithme :
  - ❑ gestion des entrées-sorties,
  - ❑ affectation d'une valeur et mise en forme d'un calcul,
  - ❑ **en opérant essentiellement sur des nombres entiers.**

# Organisation du programme:

**Le programme est divisé en trois parties:**

- Fonctions
- Géométrie
- Statistiques et probabilités

Les capacités attendues dans le domaine de l'algorithmique d'une part et du raisonnement d'autre part, sont transversales et doivent être développées à l'intérieur de chacune des trois parties.

**Des activités de type algorithmique possibles sont signalées dans les différentes parties du programme et précédées du symbole  $\diamond$ .**

# Activités des élèves

Dans le cadre de cette activité algorithmique, les élèves sont entraînés à :

- décrire certains algorithmes en langage naturel ou dans un langage symbolique ;
- en réaliser quelques uns à l'aide d'un tableur ou d'un petit programme réalisé sur une calculatrice ou avec un logiciel adapté ;
- interpréter des algorithmes plus complexes.

Aucun langage, aucun logiciel n'est imposé.

L'algorithmique a une place naturelle dans tous les champs des mathématiques et les problèmes posés doivent être en relation avec les autres parties du programme (fonctions, géométrie, statistiques et probabilité, logique) mais aussi avec les autres disciplines ou la vie courante.

# Document d'accompagnement

## **Algorithmes et démarche algorithmique**

La sensibilisation de l'élève à la question de la « démarche algorithmique » pourra se faire en évitant toute technicité ou exposé systématique. On pourra sur ce thème consulter des publications réalisées dans le cadre des IREM.

Les compétences suivantes pourront être identifiées et travaillées :

- comprendre et analyser un algorithme préexistant ;
- modifier un algorithme pour obtenir un résultat particulier ;
- analyser la situation : identifier les données d'entrée, de sortie, le traitement...;
- mettre au point une solution algorithmique : comment écrire un algorithme en « langage courant » en respectant un code, identifier les boucles, les tests, des opérations d'écriture, d'affichage... ;
- valider la solution algorithmique par des traces d'exécution et des jeux d'essais simples ;
- adapter l'algorithme aux contraintes du langage de programmation : identifier si nécessaire la nature des variables... ;
- valider un programme simple.

# Capacités attendues

**Instructions élémentaires** (affectation, calcul, entrée, sortie).

Les élèves, dans le cadre d'une résolution de problèmes, doivent être capables :

- d'écrire une formule permettant un calcul ;
- d'écrire un programme calculant et donnant la valeur d'une fonction ;
- ainsi que les instructions d'entrées et sorties nécessaires au traitement.

# Capacités attendues

## **Boucle et itérateur, instruction conditionnelle:**

Les élèves, dans le cadre d'une résolution de problèmes, doivent être capables de:

- programmer un calcul itératif, le nombre d'itérations étant donné ;
- programmer une instruction conditionnelle, un calcul itératif, avec une fin de boucle conditionnelle.

# Mon expérience personnelle

EIHassan FADILI  
Lycée Salvador Allende

# Algorithmes opératoires

EIHassan FADILI  
Lycée Salvador Allende

# Extraits de sujets du DNB

(France septembre 2008)

## Algorithme 1 :

On donne le programme de calcul suivant (algorithme) :

- a. Choisir un nombre  $x$ .
- b. Calculer le carré de ce nombre.
- c. Multiplier par 10.
- d. Ajouter 25.
- e. Afficher le résultat.

## Consignes :

$\sqrt{2}$

1. Mathieu a choisi 2 comme nombre de départ et il a obtenu 65. Vérifier par un calcul que son résultat est exact.
2. On choisit comme nombre de départ racine de 2. Que trouve-t-on comme résultat ?
3. Clémence affirme que si le nombre choisi au départ est un nombre entier pair alors le résultat est pair. A-t-elle raison ? Justifier.
4. Margot affirme que le résultat est toujours positif quelque soit le nombre choisi au départ. A-t-elle raison ? Justifier.
5. Traduire l'algorithme par une formule en fonction de  $x$ .

EIHassan FADILI

Lycée Salvador Allende

# Instruction conditionnelle

- **Si...alors**
- **Si...alors...sinon**

# Test de divisibilité par 7

## Consignes:

- 1. Rédiger un texte, avec des tâches très précises à effectuer dans un certain ordre, qui permette à votre voisin de savoir si un entier  $N$  est divisible par 7.**
- 2. Echanger votre texte avec votre voisin et exécuter exactement ce que vous dit le texte avec des entiers de votre choix.**
- 3. Corriger si nécessaire le texte de votre voisin.**

# Solutions proposées

## Solution 1:

On suppose que le voisin connaît la division euclidienne.

1. Choisir un entier  $N$
2. Calculer le reste de la division euclidienne de  $N$  par 7
3. Si  $R$  est égal à 0 alors  $N$  est divisible par 7 sinon  $N$  n'est pas divisible par 7.

## Solution 2:

On suppose que le voisin ne connaît pas la division euclidienne, mais il dispose d'une calculatrice.

1. Choisir un entier  $N$
2. Calculer  $N \div 7$
3. Si le résultat obtenu est un entier alors  $N$  est divisible par 7 sinon  $N$  n'est pas divisible par 7.

# Programme avec Algobox

(projeté en classe pour présenter le logiciel)

CODE DE L'ALGORITHME :

```
VARIABLES
  N EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
  AFFICHER "Entrer un entier N"
  LIRE N
  SI (N%7==0) ALORS
    DEBUT_SI
      AFFICHER N
      AFFICHER "est divisible par 7"
    FIN_SI
  SINON
    DEBUT_SINON
      AFFICHER N
      AFFICHER "n'est pas divisible par 7"
    FIN_SINON
FIN_ALGORITHME
```

AlgoBox

EIHassan FADILI  
Lycée Salvador Allende

# TP1 sur Algobox

(voir polycopié)

1. Test de divisibilité par 7.
2. Algorithme 1: présenté en classe avec la consigne d'afficher tous les résultats intermédiaires.
3. Algorithme 2: (même consigne que ci-dessus)
  - a. Choisir un entier naturel  $n$  qui correspond au nombre de fois où tu aimerais aller au cinéma par semaine.
  - b. Le multiplier par 2.
  - c. Ajouter 7 au résultat.
  - d. Multiplier le résultat par 50.
  - e. Si tu as déjà fêté ton anniversaire en 2009 ajouter 1659 sinon ajouter 1658.
  - f. Soustraire ton année de naissance au résultat.
  - g. Afficher le résultat.

# Boucle et itérateur

- Répéter n fois...
- Répéter ...tant que...

# Exemple 1: Vers la conjecture de Syracuse

**On considère l'algorithme suivant :**

- 1. Entrer un entier naturel  $N$ .**
- 2. Tant que  $N > 1$  répéter la procédure suivante :**
  - Si  $N$  est pair remplacer  $N$  par  $N \div 2$ .**
  - Sinon remplacer  $N$  par  $2 \times N + 1$ .**
- 3. Afficher la valeur de  $N$ .**

Consignes:

- a) Réaliser cet algorithme avec les entiers  $N = 4$ ,  $N = 3$ , puis  $N = 6$ .
- b) Que constatez-vous ?

# Exemple 2 : Conjecture de Syracuse

On considère l'algorithme suivant :

1. Entrer un entier naturel  $N$ .
2. Tant que  $N > 1$  répéter la procédure suivante :
  - Si  $N$  est pair remplacer  $N$  par  $N \div 2$ .
  - Sinon remplacer  $N$  par  $3 \times N + 1$ .
3. Afficher la valeur de  $N$ .

Consignes: (travail à la main!)

- a) Réaliser cet algorithme avec les entiers  $N = 6$ ,  $N = 7$ , puis  $N = 16$ .
- b) Que constatez-vous ?
- c) Modifier l'algorithme pour qu'il affiche toutes les valeurs successives de  $N$ .
- d) Modifier l'algorithme pour qu'il affiche le nombre de tests effectués.
- e) Modifier l'algorithme pour qu'il affiche la valeur maximale de  $N$  atteinte.

# TP2 sur Algobox

## Consignes: (avec $3N + 1$ )

1. Réaliser un programme qui réalise l'algorithme initial(Syracuse0).
2. Tester le programme avec des entiers de votre choix.
3. Modifiez le programme pour qu'il affiche à chaque étape la nouvelle valeur de N et tester à nouveau le programme(Syracuse1).
4. Modifiez le programme pour qu'il affiche le nombre d'itérations et tester à nouveau le programme(Syracuse2).
5. Modifiez le programme pour qu'il affiche le nombre maximal atteint et tester à nouveau le programme(Syracuse3).
6. Prolongement possible: (algorithme de tri)  
Modifiez le programme pour qu'il affiche la liste de tous les nombres calculés dans **l'ordre croissant**.

# TP: limites d'un programme!

(plus tard dans l'année)

## Consignes: (avec $2N + 1$ )

1. Réaliser un programme.
2. Tester le programme avec des nombres qui ne sont divisibles que par 2 et avec d'autres entiers.
3. Que constatez-vous ?
4. Modifiez le programme pour qu'il affiche à chaque étape la nouvelle valeur de N.
5. Testez à nouveau le programme.
6. Expliquez ce que vous observez.

# Conjecture de Syracuse sur Excel

N	$n/2$ ou $3n + 1$
29	88
88	44
44	22
22	11
11	34
34	17
17	52
52	26
26	13
13	40
40	20
20	10
10	5
5	16
16	8
8	4
4	2
2	1
1	<b>FIN</b>

N	$n/2$ ou $3n + 1$
7	22
22	11
11	34
34	17
17	52
52	26
26	13
13	40
40	20
20	10
10	5
5	16
16	8
8	4
4	2
2	1
1	<b>FIN</b>

# Syracuse sur TI-82

The image shows a TI-82 calculator screen with a green background and black text. At the top left, there is a small Texas Instruments logo and the text "TEXAS INSTRUMENTS". At the top right, it says "TI-82". The main screen area displays the following code:

```
PROGRAM: CYRACUSE
:Disp "N#1"
:Prompt N
:While N#1
:If int (N/2)=N/
2
:Then
:N/2→N
:Else
:3*N+1→N
:End
:End
:Disp N
:
```

# Code du programme de Syracuse sur VBA

```
Public Sub Syracuse()
```

```
Dim N, v, k, max As Double
```

```
N = InputBox("entrer un entier positif autre que 1")
```

```
v = N
```

```
k = 0
```

```
max = N
```

```
While v <> 1
```

```
    If v Mod 2 = 0 Then
```

```
        v = v / 2
```

```
    Else
```

```
        v = 3 * v + 1
```

```
    End If
```

```
    k = k + 1
```

```
    If v / max > 1 Then
```

```
        max = v
```

```
    End If
```

```
Wend
```

```
MsgBox "Pour l'entier " & N & " Syracuse a nécessité " & k & " etapes"
```

```
MsgBox "et le nombre le plus haut atteint est : " & max
```

```
End Sub
```

### Exemple 3: produit de deux entiers positifs.

1. Entrer deux entiers positifs a et b.
2. Initialiser les variables p à 0.
3. Répéter b fois :
  - Remplacer p par  $p + a$ .
4. Afficher p.

Réaliser cet algorithme avec quelques entiers de votre choix. Que calcule cet algorithme ?

# Programme sur TI 82

The image shows a TI-82 calculator screen with a program named 'MULTIPLI'. The screen displays the following code:

```
TEXAS INSTRUMENTS TI-82
PROGRAM: MULTIPLI
: Prompt A
: Prompt B
: 0 → P
: 1 → K
: While K ≤ B
: P + A → P
: K + 1 → K
: End
: Disp P
:
```

# Code du programme sur VBA

```
Public Sub MultipliDeuxEntiers()
```

```
Dim a, b, p, k As Integer
```

```
a = InputBox("entrer un entier positif a")
```

```
b = InputBox("entrer un entier positif b")
```

```
p = 0
```

```
k = 1
```

```
While k <= b
```

```
    p = p + a
```

```
    k = k + 1
```

```
Wend
```

```
MsgBox "le produit de " & a & " par " & b & " est  
    égal à " & p
```

```
End Sub
```

# Algorithmes en arithmétique (quasi hors programme!)

EIHassan FADILI  
Lycée Salvador Allende

1. Pgcd par soustractions successives
2. Calcul du quotient et reste d'une division euclidienne
3. Algorithme d'Euclide
4. Liste des diviseurs d'un entier
5. Test de primalité (avec ou sans le test racine de N)
6. Crible d'Eratosthène
7. Factorisation d'un entier

# Algorithmes en algèbre

résolution équation second degré  
(forme canonique)

# Algorithmes en analyse

- Dichotomie
- Extrêêmes d'une fonction
- Sens de variation d'une fonction

# Algorithmes en probabilités- statistiques

- Calcul d'une moyenne, d'une médiane(tri)
- Gain d'un joueur (espérance!)
- Lièvre tortue (simulation)

# Gain du joueur

Un joueur lance deux dés et fait la somme des points obtenus. S'il obtient 8, il gagne 10€, sinon il perd 1€.

## Variante 1:

Le joueur rejoue 10 fois (et cumule ses gains et pertes).

## Variante 2:

Le joueur rejoue jusqu'à avoir un gain cumulé de 5€.

# Le lièvre et la tortue

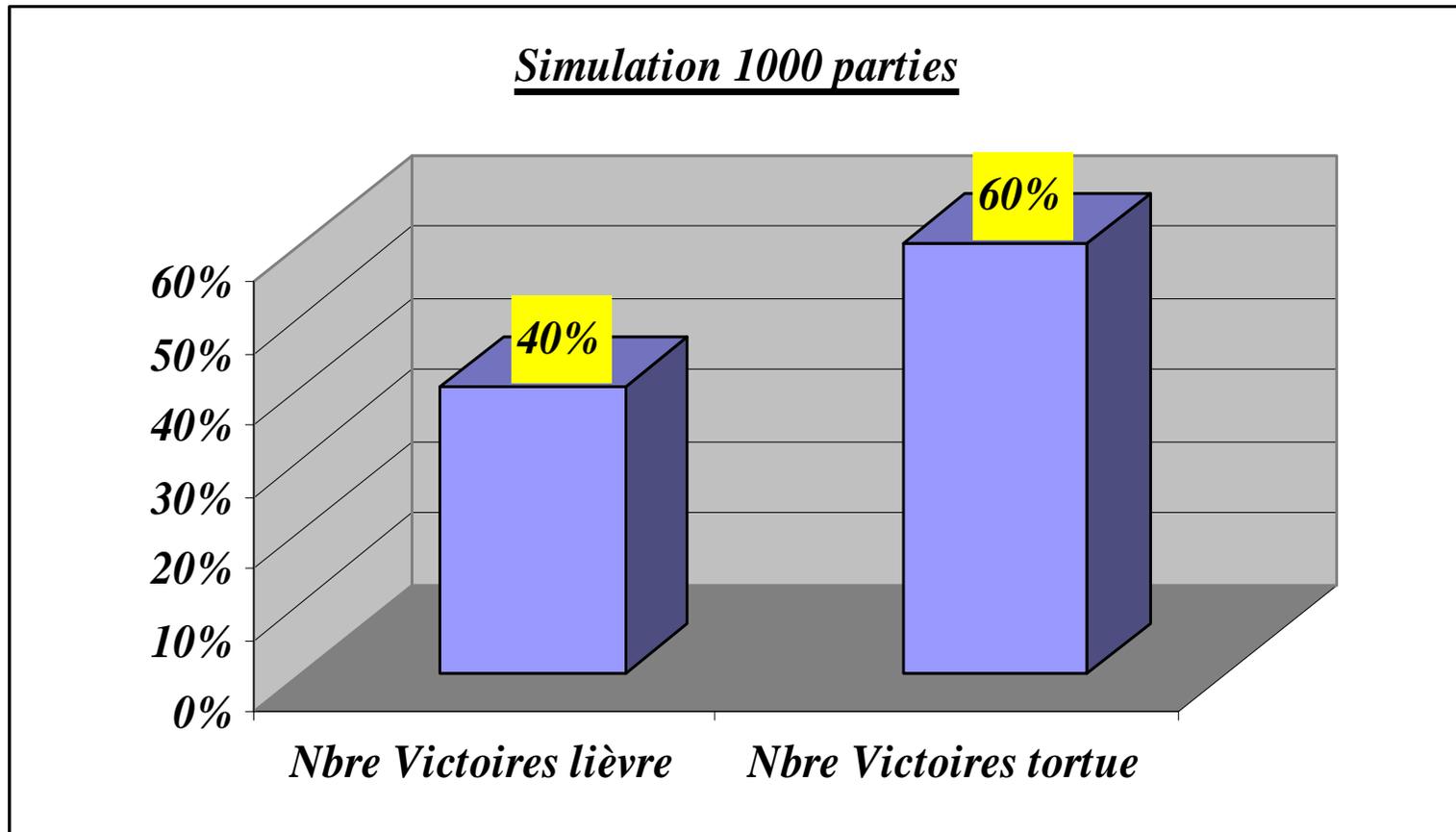
Le lièvre est plus rapide que la tortue.

Pour donner plus de chance à la tortue on adopte la règle de jeu suivante:

- Un parcours est composé de 5 cases
- On lance un dé non pipé à 6 faces
- Si on obtient un chiffre autre que 6 la tortue avance d'une case
- Sinon c'est le lièvre qui a gagné la course
- Continuer jusqu'à la victoire du lièvre ou de la tortue

A ce jeu lequel des deux gagnera le plus souvent ?

# Simulation sur Excel



# Algorithmes en Géométrie

EIHassan FADILI  
Lycée Salvador Allende

# Exemples

- Calcul des coordonnées du milieu d'un segment
- Calcul d'une distance
- Nature d'un triangle

# EVALUATION

EiHassan FADILI  
Lycée Salvador Allende

# Exemple d'épreuve bac L (France juin 2007)

On considère l'algorithme suivant :

- **Entrée** :  $a$  un entier naturel.
- **Initialisation** :  $L$  liste vide
- Affecter la valeur  $a$  à  $x$ .
- **Traitement** : Tant que  $x > 0$  ;
  - Effectuer la division euclidienne de  $x$  par 7 ;
  - Affecter son reste à  $r$  et son quotient à  $q$  ;
  - Mettre la valeur de  $r$  au début de la liste  $L$  ;
  - Affecter  $q$  à  $x$ .
- **Sortie** : Afficher les éléments de la liste  $L$ .

# Exemple d'épreuve bac L (France juin 2007)

1. Faire fonctionner cet algorithme pour  $a = 486$ . On reproduira sur la copie un tableau analogue à celui donné ci-dessous et on le complétera.
2. Expliquer le lien entre les éléments de la liste  $L$  et l'écriture de  $u_5$  en base 7.

	$r$	$q$	$L$	$x$
Initialisation			vide	486
Fin étape 1				
Fin étape 2				
...				
...				
...				

END.....merci

EiHassan FADILI  
Lycée Salvador Allende