


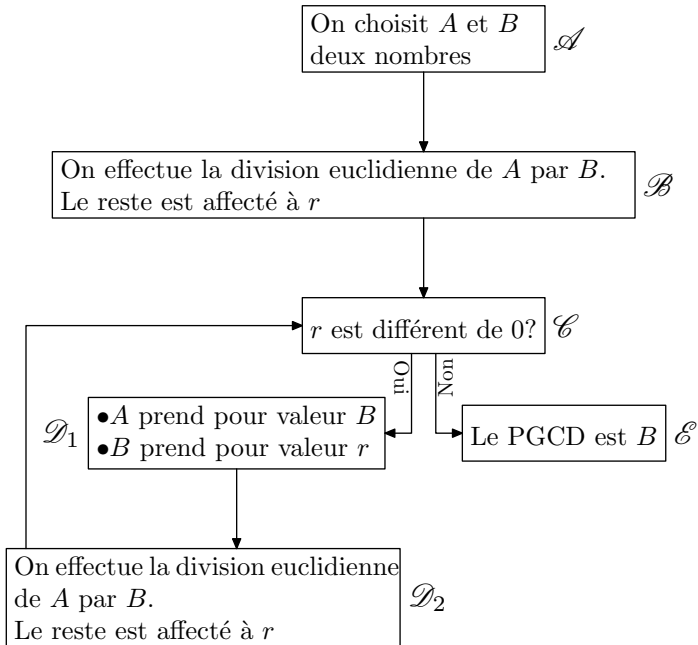


# Seconde / Algorithmes

## 1. Etude générale d'algorithmes

**E.1**    Le diagramme ci-dessous représente l'algorithme d'Euclide déterminant le plus grand diviseur commun de deux nombres entiers :






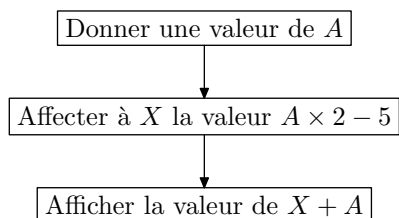
|                 |                            |
|-----------------|----------------------------|
| $\mathcal{A}$   | $A = 254$ et $B = 16$      |
| $\mathcal{B}$   | $r = 14$                   |
| $\mathcal{C}$   | Oui                        |
| $\mathcal{D}_1$ | $A = 16$ et $B = 14$       |
| $\mathcal{D}_2$ | $r = 2$                    |
| $\mathcal{C}$   | Oui                        |
| $\mathcal{D}_1$ | $A = 14$ et $B = 2$        |
| $\mathcal{B}$   | $r = 0$                    |
| $\mathcal{C}$   | Non                        |
| $\mathcal{D}_2$ | $\text{PGCD}(254; 16) = 2$ |

① En reproduisant de manière analogue le tableau ci-dessous, déterminer le *PGCD* des entiers suivants :

a  $A = 1542$  ;  $B = 36$       b  $A = 18$  ;  $B = 543$




② Pour les valeurs de la question b, qu'effectue l'algorithme au début de cet algorithme?

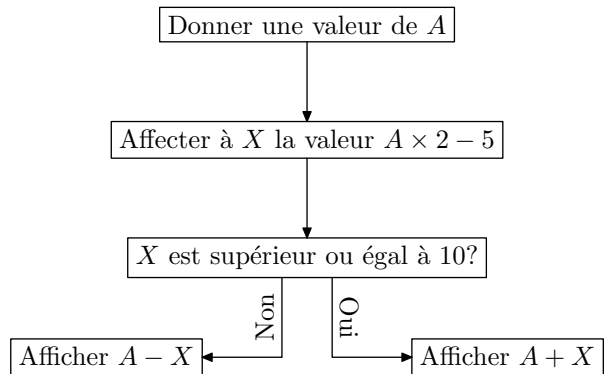
**E.2**    On considère l'algorithme dans la représentation est donnée par le graphique ci-dessous :



Compléter le tableau ci-dessous :




|                 |   |   |    |               |    |
|-----------------|---|---|----|---------------|----|
| Valeur de A     | 0 | 3 | 12 | $\frac{5}{2}$ | -4 |
| Valeur affichée |   |   |    |               |    |

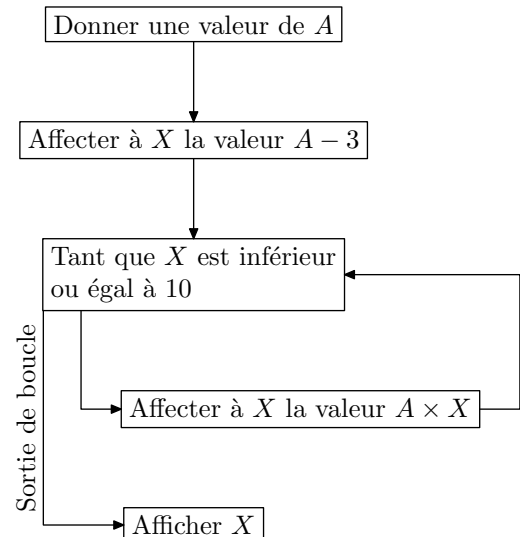
**E.3**    On considère l'algorithme dont la représentation est donnée par le graphique ci-dessous :



Compléter le tableau ci-dessous :

|                 |   |   |    |   |    |
|-----------------|---|---|----|---|----|
| Valeur de A     | 5 | 8 | -2 | 0 | 21 |
| Valeur affichée |   |   |    |   |    |

**E.4**    On considère l'algorithme dans la représentation est donnée par le graphique ci-dessous :



① Justifier qu'en affectant la valeur 5 à la variable A, l'algorithme affiche la valeur 50?

② Déterminer la valeur affichée par l'algorithme dans les cas suivants :

a  $A = 10$       b  $A = 4$       c  $A = -2$

③ a Que se passe-t-il lorsqu'on affecte la valeur 0 à la variable A?

b Trouver un autre exemple où l'algorithme ne se termine jamais.

## 2. Première utilisation d'algoBox

E.5   

- 1 La commande `floor` permet d'obtenir la partie entière d'un nombre; supposons que la variable `a` ait la valeur 3,1415926535.

- a Donner la valeur de `floor(a*10)`.  
b En déduire la commande pour obtenir la valeur par défaut de `a` au dixième près; au centième près.

- 2 Déterminer les restes des divisions euclidienne suivante:

a 10 par 3      b 33 par 5

c 27 par 4      d 69 par 8

La commande `a%b` renvoie le reste de la division euclidienne de `a` par `b`:

- e Quelles peuvent être les valeurs de `a%2`? de `2%a`?  
f À l'aide d'une structure conditionnelle, écrire un algorithme demandant la saisie d'une valeur puis qui affiche les phrases "ce nombre est pair" ou "ce nombre est impair" suivant les cas.

- 3 La commande `sqrt(2)` renvoie la racine carrée du nombre 2:

- a Écrire un algorithme demandant à l'utilisateur quatre nombres représentant les coordonnées de deux points, et renvoyant la distance séparant ces deux points.  
b Modifier l'algorithme pour qu'il affiche la valeur par défaut de cette distance au dixième près.

E.6   

- 1 Saisir l'algorithme ci-dessous dans le langage de programmation de votre choix:

```
Pour a allant de 1 à 25
  x ← a × a
Fin Pour
```

- 2 Par une exécution pas à pas de cet algorithme, donner l'ensemble des valeurs qui seront affectées à la variable `x`.

E.7   

- 1 Saisir l'algorithme ci-dessous dans le langage de programmation de votre choix:

```
a ← 0
Tant que a < 100
  x ← a%2
  Si x = 0
    Alors
      y ← a
    Fin Si
  a ← a+1
Fin Tant que
```

- 2 Lors de l'exécution de l'algorithme pas à pas, quelles sont les différentes valeurs affectées à la variable `y`.  
3 Modifier cet algorithme afin que la variable `y` soit affectée successivement de tous les multiples de 13 inférieurs à 100.




E.8   

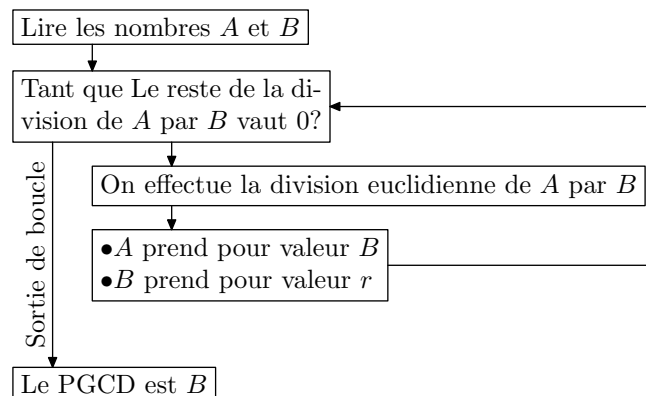
- 1 Saisir l'algorithme ci-dessous dans le langage de programmation de votre choix:

```
Fonction f(a)
  a ← √a
  Tant que a >= 1
    a ← a-1
  Fin Tant
  Renvoyer a
```




- 2 a Effectuer un appel à la fonction `f` avec chacune des valeurs suivantes:  
1 ; 2 ; 3 ; 4 ; 9 ; 10  
b Quel est le rôle de la fonction `f`?

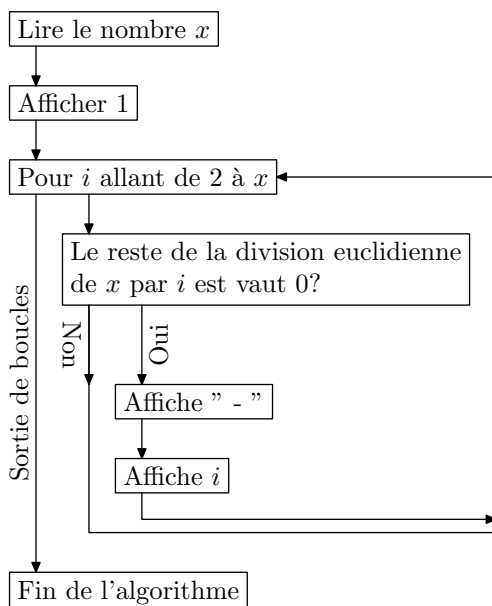
## 3. Création d'algorithme

E.9    Le schéma ci-dessous représente l'algorithme d'Euclide.



Construire cet algorithme à l'aide d'algoBox.

E.10    On considère l'algorithme ci-dessous où les variables `x` et `i` sont de type nombre:



- ① Construire cet algorithme à l'aide d'Algobox.
- ② Mathématiquement, à quoi sert cet algorithme?

#### 4. Tracé de courbes

E.11

- ① a Dans AlgoBox et dans l'onglet "Dessiner dans un repère", cocher la case "Utiliser un repère"; saisir les valeurs suivantes pour les bornes des axes :

Xmin:-5 ; Xmax:5 ; GraduationsX:1

Ymin:0 ; Ymax:25 ; GraduationsY:1

- b Saisir dans AlgoBox l'algorithme ci-dessous :

```

▼VARIABLES
  x EST_DU_TYPE NOMBRE
  y EST_DU_TYPE NOMBRE
  i EST_DU_TYPE NOMBRE
▼DEBUT_ALGORITHME
  ▼POUR i ALLANT DE -5 A 5
    DEBUT_POUR
    x PREND_LA_VALEUR i
    y PREND_LA_VALEUR x*x
    TRACER_POINT (x,y)
    FIN_POUR
  FIN_POUR
FIN_ALGORITHME
  
```

- c Exécuter l'algorithme pour observer son affichage.
- d Que semble afficher cet algorithme?
- 2 On souhaite tracer plus de points représentant cette courbe, pour cela, on souhaite modifier la boucle itérative pour que les abscisses des points soient espacées de 0,1 en 0,1 :
  - a Modifier la ligne `x PREND_LA_VALEUR i` en :  
`x PREND_LA_VALEUR i/10`
  - b Exécuter l'algorithme pour observer l'effet de ces modifications.
  - c Quelle modification faut-il effectuer sur l'algorithme pour que la courbe représentative s'affiche sur l'intervalle  $[-5; 5]$ ?
  - d Appliquer ces changements et relancer cet algorithme.

E.12 On considère la fonction  $f$  définie par :

$$f(x) = \sqrt{x-1}$$

Dans un repère  $(O; I; J)$  orthogonal, on note  $\mathcal{C}_f$  la courbe représentative de la fonction  $f$ .

- ① a Déterminer l'ensemble de définition de la fonction  $f$ .
- b Déterminer les coordonnées du point de la courbe  $\mathcal{C}_f$  ayant pour abscisse 1.
- 2 a Dans AlgoBox et dans l'onglet "Dessiner dans un repère", cocher la case "Utiliser un repère"; saisir les valeurs suivantes définissant les bornes des axes :

Xmin:0 ; Xmax:9 ; GraduationsX:1

Ymin:0 ; Ymax:3 ; GraduationsY:1

- b Saisir dans AlgoBox l'algorithme suivant :

```

▼VARIABLES
  x1 EST_DU_TYPE NOMBRE
  y1 EST_DU_TYPE NOMBRE
  x2 EST_DU_TYPE NOMBRE
  y2 EST_DU_TYPE NOMBRE
  i EST_DU_TYPE NOMBRE
▼DEBUT_ALGORITHME
  ▼POUR i ALLANT DE 10 A 100
    DEBUT_POUR
    x2 PREND_LA_VALEUR i/10
    y2 PREND_LA_VALEUR sqrt(x2-1)
    TRACER_POINT (x2,y2)
    FIN_POUR
  FIN_POUR
FIN_ALGORITHME
  
```

- c Exécuter l'algorithme et observer le graphique obtenu.
- 3 Le but de cette question est de tracer la courbe  $\mathcal{C}_f$  par des segments reliant chacun des points précédents.
  - a Effacer la commande "TRACER\_POINT (x2,y2)" pour la remplacer par la commande TRACER\_SEGMENT reliant les points de coordonnées (x1,y1) et (x2,y2).

- (b) Exécuter l'algorithme pour observer les modifications. Le tracé effectué est composé uniquement par des segments. Quel est l'origine commune à tous ces segments? Pourquoi?
- (c) Avant la définition de la boucle **for** et en relation avec la question (1) (b), initialiser correctement les valeurs de  $x_1$  et  $y_1$  afin d'améliorer le tracé de la courbe  $\mathcal{C}_f$ .

- (d) Pour tracer la courbe  $\mathcal{C}_f$  segment par segment, l'algorithme doit relier le point actuel avec le point précédent. Juste avant la fin de la boucle **POUR**, faire en sorte que le point de coordonnées  $(x_1, y_1)$  représente le point de coordonnées  $(x_2, y_2)$  pour la prochaine exécution de la boucle.

## 5. Observation de la loi des grands nombres

E.13   

- (1) (a) Saisir dans l'algorithme de votre choix l'algorithme suivant:

```

c ← 0
Pour i allant de 0 à 100
    x ← valeur aléatoire appartenant à [0 ; 1[
    x ← partie entière de 3 × x
Fin Pour

```

- (b) En exécutant pas à pas cet algorithme, quelles sont les valeurs affectées à la variable "x"?
- (2) (a) Ajouter une structure conditionnelle à l'intérieur de la boucle **POUR** afin que l'instruction:




```
c ← c+1
```

soient exécutés à chaque fois que la variable "x" soit affecté de la valeur 2.

- (b) Exécuter plusieurs fois l'algorithme et observer la valeur de la variable c. Peut-on expliquer les variations des valeurs de la variable c?
- (3) (a) Modifier l'algorithme afin que la boucle effectue 500 itérations et ajouter l'instruction ci-dessous en fin d'algorithme:

```
f ← c / 500
```

- (b) Exécuter plusieurs fois cet algorithme et observer les variations de la valeur de la variable f en fin d'algorithme.
- (4) Que peut-on faire pour que les variations de la variable f se stabilise?

E.14   

- (1) (a) Activer l'utilisation d'un repère dans AlgoBox en prenant les paramètres suivant:

Xmin: 0    Xmax: 10    Graduations X: 1

Ymin: 0    Ymax: 10    Graduations Y: 1

- (b) Saisir l'algorithme suivant dans AlgoBox:

```

▼VARIABLES
  | x EST_DU_TYPE NOMBRE
  | i EST_DU_TYPE NOMBRE
  | max EST_DU_TYPE NOMBRE
▼DEBUT_ALGORITHME
  | max PREND_LA_VALEUR 100
  | ▼POUR i ALLANT DE 1 A 100
  |   | DEBUT_POUR
  |   | x PREND_LA_VALEUR random()
  |   | x PREND_LA_VALEUR floor(x*3)
  |   | TRACER_POINT (10*i/max, x)
  |   | FIN_POUR
  | FIN_ALGORITHME

```

- (c) Exécuter cet algorithme. Quelle est son action?
- (2) (a) En utilisant l'exercice précédent, modifier l'algorithme présent pour qu'il affiche la fréquence d'apparition du nombre 2 (dans la variable x).
- (b) Modifier l'algorithme pour qu'il affiche la droite d'équation  $y=1/3$ .
- (c) Augmenter le nombre de tirages de cet algorithme. Quelle observation peut-on faire lors de l'exécution de l'algorithme?

## 6. Dichotomie

E.15   

- (1) Saisir l'algorithme ci-dessous:

```

▼VARIABLES
  | borneMin EST_DU_TYPE NOMBRE
  | borneMax EST_DU_TYPE NOMBRE
  | x EST_DU_TYPE NOMBRE
▼DEBUT_ALGORITHME
  | ▼TANT_QUE (borneMax-borneMin>pow(10,-3)) FAIRE
  |   | DEBUT_TANT_QUE
  |   | ▼SI (x<(borneMin+borneMax)/2) ALORS
  |   |   | DEBUT_SI
  |   |   | borneMax PREND_LA_VALEUR (borneMin+borneMax)/2
  |   |   | FIN_SI
  |   | ▼SINON
  |   |   | DEBUT_SINON
  |   |   | borneMin PREND_LA_VALEUR (borneMax+borneMin)/2
  |   |   | FIN_SINON
  |   | AFFICHER borneMin

```

- 2 a Exécuter cet algorithme avec les valeurs suivantes :  
xMin=1 ; borneMax=3 ; x=1.9384
- b En observant les valeurs successives prises par borneMin et borneMax, vers quelle valeur les nombres

borneMin et borneMax se dirigent-ils?

- 3 Modifier cet algorithme pour que ces deux valeurs se rapprochent de  $\sqrt{2}$ .