

C3T3 – Algorithmique, tableur et pgcd

Partie 1 Algorithmique et tableur (à faire en classe entière)

1. Définition

Un algorithme est une suite finie et non-ambiguë d'instructions permettant de donner la réponse à un problème. [Le mot « algorithme » vient du nom du mathématicien Al Khuwarizmi (latinisé au Moyen Âge en *Algoritmi*)]

2. Quelques explications sur un exemple

L'instruction 1 signifie : « Choisir 2 nombres entiers, écrire le plus grand dans une case appelée A et le plus petit dans une case appelée B »

Les instructions 2 et 3 sont des tests. « Est-ce que le contenu de la case B est nul ? ». Suivant la réponse on prend le chemin « yes » ou le chemin « no ».

« Est-ce que le contenu de la case A est plus grand que le contenu de la case B ? ». Suivant la réponse on prend le chemin « yes » ou le chemin « no ».

L'instruction 4 signifie : « Soustraire le contenu de la case A à celui de la case B, puis (flèche) écrire le résultat dans la case B »

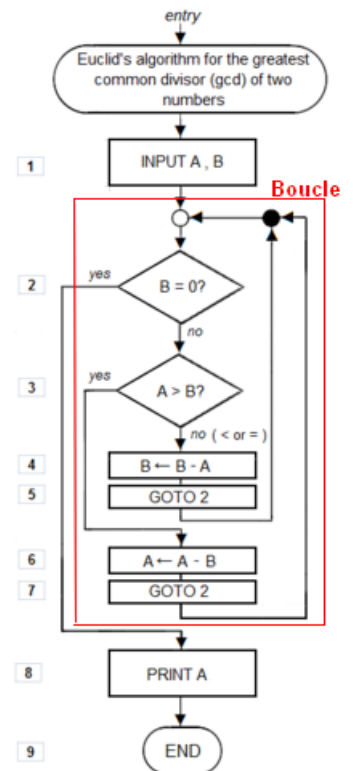
L'instruction 5 signifie : Retourner à l'instruction 2.

L'instruction 6 signifie : « Soustraire le contenu de la case B à celui de la case A, puis (flèche) écrire le résultat dans la case A »

L'instruction 7 signifie : Retourner à l'instruction 2.

L'instruction 8 signifie « annoncer que le pgcd est le contenu de la case A ».

L'instruction 9 : on a terminé.



Remarques : Le fait de retourner sur l'instruction 2 constitue une boucle. On en sort quand le test donne « valeur de B nulle ». Chaque exécution de la boucle s'appelle une étape. L'algorithme comprend un nombre fini d'étapes puisque les contenus des cases A et B décroissent à chaque exécution de la boucle : on est bien en présence d'un algorithme.

- Faisons tourner cet algorithme « à la main » en choisissant 8 comme contenu de la case A et 6 contenu de la case B. (Tourner à la main signifie faire à la place de l'ordinateur).
- Le fait d'effacer et de réécrire dans les cases n'étant pas très pratique tourner à nouveau l'algorithme « à la main » en utilisant le tableau ci-dessous et en choisissant comme valeurs de départ 6 pour A et 9 pour B.

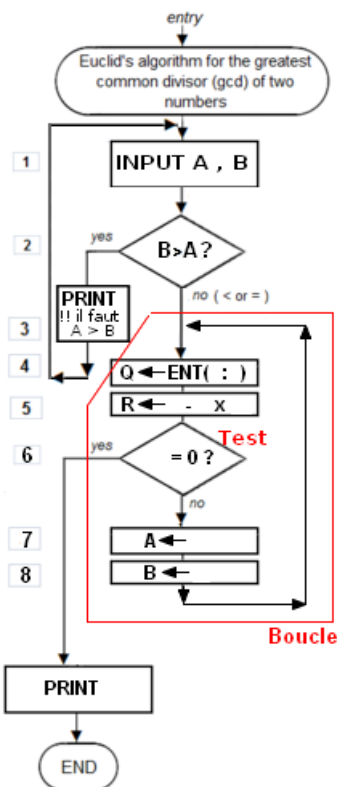
	Contenu de la case A	Contenu de la case B
Valeurs initiales	6	9
Valeurs à la fin de l'étape 1		
Valeurs à la fin de l'étape 2		
Valeurs à la fin de l'étape 3		
Résultat		

- Refaites tourner l'algorithme avec 2 nombres de votre choix et présenter les résultats sous forme de tableau.

C3T3 – Algorithmique, tableur et pgcd

3. Algorithme d'Euclide

Compléter l'algorithme d'Euclide ci-dessous, puis le « tourner à la main » avec les nombres 472 et 77 en complétant le tableau proposé à côté.



	a	b	q	r
Valeurs initiales	472	77	XXX	XXX
Valeurs avant le test de l'étape 1	472	77		
Valeurs avant le test de l'étape 2				
Valeurs avant le test de l'étape 3				
Valeurs avant le test de l'étape 4				
Valeurs avant le test de l'étape 5				

Correction [ici](#)

4. Tableur

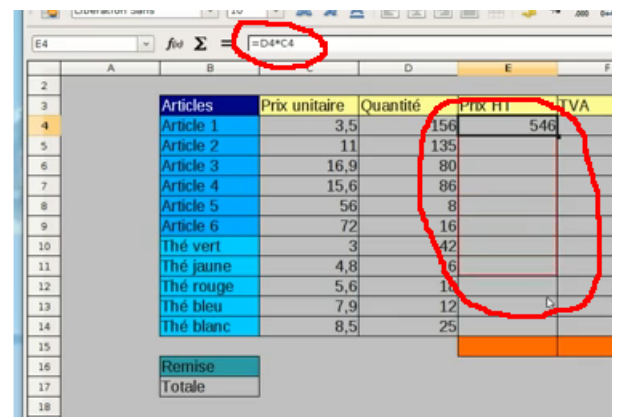
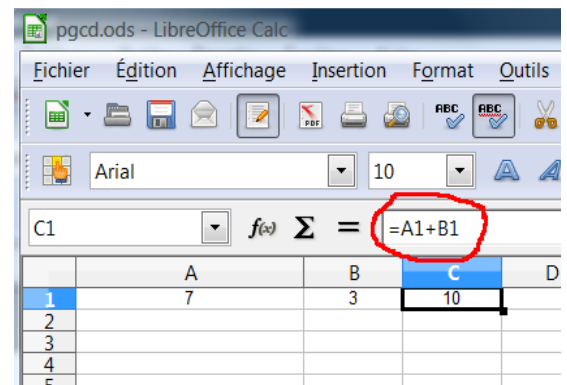
Un tableur est un outil, inclus dans une suite bureautique, qui permet de créer des tableaux.

Dans chaque case, appelée « cellule », on peut écrire directement un contenu. Exemple : « 7 » dans la case A1, « 3 » dans la case B1.

Mais on peut également demander à l'ordinateur d'évaluer une expression et d'écrire le résultat du calcul dans une case déterminée. Exemple : Le fait de saisir « =A1+B1 » dans la case C1 provoque l'affichage de la somme 10. (Ne pas oublier le signe =)

L'intérêt du tableur, c'est qu'il met automatiquement à jour le contenu des cases contenant une expression. Si, par exemple, on modifie le contenu de la case A1 pour y mettre 5, la case C1 affichera immédiatement 8.

Un outil puissant du tableur est le fait de permettre de recopier une ligne (ou une partie de ligne) en modifiant automatiquement les références des cellules concernées. Dans l'exemple ci-contre, en cliquant sur E4 et en tirant vers le bas (maintenir le bouton gauche enfoncé) on recopie la formule qui écrit le produit D5*C5 en E5 ; D6*C6 en E6 ... Voir le petit [tutoriel en video ici](#).



C3T3 – Algorithmique, tableur et pgcd

Partie 2 Préparation à la main (à faire en classe par groupe)

1. Un exemple : Calcul du PGCD de 472 et 77

a. Complète le tableau suivant en utilisant l'algorithme d'Euclide (par divisions successives) :

	A	B	C (quotient)	D (reste)
Étape 1 (Valeurs initiales)	472	77		
Étape 2				
Étape 3				
Étape 4				
Étape 5				

b. Analyse du tableau :

Pour nommer une case du tableau, on utilise le nom de sa colonne et de sa ligne :

Exemple : A1 correspond à la case de la colonne A, ligne « Étape 1 » : elle contient 472.

- Quel nombre y a-t-il dans la case B1 ? dans la case A2 ?

Pour les questions suivantes, répondre en **utilisant les noms** des cases, et non pas les valeurs :

Exemple : Dans la case C1, il y a le quotient de la division de A1 par B1.

- Que doit-il y avoir dans la case D1 ?
- Que doit-il y avoir dans la case A2 ?
- Que doit-il y avoir dans la case B2 ?
- Que doit-il y avoir dans la case C2 ?
- Que doit-il y avoir dans la case D2 ?
- Quel est le numéro de la dernière étape ?
- Quelle est l'adresse de la case qui contient le résultat ?

2. Préparation du passage au tableur :

Voici une liste partielle de fonctions proposées par le tableur LibreOffice :

- =C2 : place la valeur de la case C2 dans la case courante.
- =A1-B1 : place la différence entre le contenu de la case A1 et de la case B1 dans la case courante.
- =QUOTIENT(C3;A4) : place le quotient de la division euclidienne du contenu de C3 par le contenu de A4.
- =A7*5 : place le produit du contenu de la case A7 par 5.
- =MOD(D3;A4) : place le reste de la division euclidienne du contenu de D3 par le contenu de A4.
- =MIN(A3;B4) : place la plus petite des valeurs contenues dans les cases A3 et B4.

En utilisant ces fonctions, complète le tableau suivant :

	A	B	C (quotient)	D (reste)
Étape 1 (Valeurs initiales)	472	77		
Étape 2				
Étape 3				

C3T3 – Algorithmique, tableur et pgcd

Partie 3 Programmation (à faire par binôme en salle informatique ou à la maison)

1. Algorithme d'Euclide

L'objectif est de créer une feuille de calcul donnant le pgcd de deux nombres a et b (avec $a < b$) à l'aide de l'algorithme d'Euclide.

- a. Lance un tableur (pas trop fort, et de préférence celui de LibreOffice) puis reproduis le tableau préparé dans la partie 2-2 (10 lignes au total). Aide : on pourra sélectionner les 4 cases de la ligne « étape 2 » et en tirant vers le bas (maintenir le bouton gauche enfoncé) recopier la programmation des cases en poursuivant sur 8 lignes.

Voici un exemple partiel de ce que l'on doit obtenir.

	A	B	C	D
1	472	77	6	10
2				
3				
4				
5	3	1	3	0
6	1	0	#DIV/0!	#VALEUR !
7	0	#VALEUR !	#VALEUR !	#VALEUR !
8	#VALEUR !	#VALEUR !	#VALEUR !	#VALEUR !
9	#VALEUR !	#VALEUR !	#VALEUR !	#VALEUR !
10	#VALEUR !	#VALEUR !	#VALEUR !	#VALEUR !
11				

- b. Comment expliquer le contenu des lignes 6 et suivantes ?
- c. En modifiant les valeurs initiales (cellules A1 et B1), détermine les pgcd suivants :
- $\text{pgcd}(452;24) = \dots$; $\text{pgcd}(225;75) = \dots$; $\text{pgcd}(113,101) = \dots$
- d. Observe et explique ce qui se passe si on entre 55 pour a et 56 pour b.
- e. Explique ce qui se passe si on entre 377 pour A et 233 pour B. A-t-on assez de place ? Que faut-il faire pour éviter cela ? Modifie ta feuille de calcul pour pouvoir obtenir le pgcd cherché.
- f. Mets en forme les cellules de façon à obtenir une belle présentation. Tu peux aussi Insérer des lignes (au dessus) et des colonnes (devant) pour rajouter un titre, des libellés et une légende.

2. Algorithme des soustractions successives (en option mais vivement conseillé)

- a. Utiliser le tableur pour obtenir le pgcd de deux nombres par la méthode des soustractions successives.

Pour programmer le tableau vu en synthèse on se sert des instructions MIN(...,...) et MAX(...,...) qui donnent respectivement le plus petit et le plus grand de deux nombres.

Pour programmer le tableau vu dans la partie 1, on utilise la commande « Si ». Exemple : =SI(A1>5;100;"trop petit") Si la valeur en A1 est supérieure à 5, la valeur 100 est inscrite dans la cellule active ; sinon, le texte "trop petit" (sans guillemets) est inscrit.

- b. Comparer les deux algorithmes (nombre d'étapes, tests valeurs initiales, complexité de la mise en œuvre ...)