



TP4 : Itérations et paramétrage (1/2)

1 ADN sous PowerPoint

Le but de cette partie est la réalisation du dessin de la molécule d'ADN telle qu'elle a été présentée lors du TD4.

- Récupérer TP04.zip avec Internet Explorer ou Firefox à l'URL : <http://membres-lig.imag.fr/dubousquet/INF112/TP04.zip>
- Sauvegardez et décompressez le fichier d'archive dans votre répertoire INF112/ADN/.
- Ouvrez ADN.ppt.

Le document contient trois macros. Il faut les activer lors de l'ouverture du document si cela est demandé.

- SucrePhosphate
- LiaisonAT
- LiaisonCG

Exercice 1.1 : Paramétrage

Paramétrez les 3 macros comme présenté en TD.

Exercice 1.2 : Une ligne d'ADN

Créez une action ADN, non paramétrée, qui permet de dessiner une ligne d'ADN aux coordonnées (100,100). On introduira un choix aléatoire pour le choix du type de liaison (voir TD4).

Exercice 1.3 : Itération

Complétez et modifiez l'action ADN pour faire en sorte que 6 lignes soient dessinées (= introduire une itération). Vérifiez que la liaison se réalise bien de façon aléatoire.

2 Suite de Fibonacci (Examen 2eme session 2004-05)

Dans le roman de Dan Brown « Da Vinci Code », on peut lire que deux nombres consécutifs de la suite de Fibonacci ont un rapport proche du nombre d'Or (approximé ici par 1,618). La suite de Fibonacci est définie ainsi :

$$U_0 = 0$$

$$U_1 = 1$$

$$U_n = U_{n-1} + U_{n-2}$$

Ainsi $U_2 = 1+0 = 1$; $U_3 = 1+1=2$; $U_4 = 2+1 = 3$; etc...

Dans la suite on se propose d'étudier un programme pour Excel, permettant de calculer le rapport entre 2 nombres consécutifs de la suite de Fibonacci pour les 10 premières valeurs.

Dans une feuille de calcul Excel, on utilisera la première colonne pour calculer les nombres de la suite. Ainsi, dans la cellule(1,1) on affichera la valeur de U_0 , dans la cellule(2,1) la valeur de U_1 , et ainsi de suite jusqu'à obtenir les 10 premières valeurs.

Dans la seconde colonne, on calculera le rapport entre deux nombres consécutifs. Ainsi, dans la cellule (i,2) on affichera la valeur du rapport entre les valeurs des cellule(i,1) et (i-1,1). Sur la troisième colonne, on calculera la différence entre le rapport calculé et le nombre d'Or.

Exercice 2.1 : Algorithme

Complétez l'algorithme ci-après, pour faire ces calculs.

```

Action Fibol0
1. Début
2. i : entier {compteur}
3.
4. {initialisation}
5. _____
6. _____
7.
8. {itération}
9. Pour i = _____ jusqu'à _____ faire
10. Cellule(i, 1) = _____
11. Fin Pour
12. Pour i = _____ jusqu'à _____ faire
13. Cellule(i, 2) = _____
14. Cellule(i, 3) = Cells(i, 2) - 1.618
15. Fin Pour
16.
17. Fin action

```

	A	B	C	D
1	0			
2	1			
3	1	1	-0,618	
4	2	2	0,382	
5	3	1,5	-0,118	
6	5	1,66666667	0,04866667	
7	8	1,6	-0,018	
8	13	1,625	0,007	
9	21	1,61538462	-0,00261538	
10	34	1,61904762	0,00104762	
11	55	1,61764706	-0,00035294	
12	89	1,61818182	0,00018182	
13	144	1,61797753	-2,2472E-05	
14	233	1,61805556	5,5556E-05	
15	377	1,61802575	2,5751E-05	
16	610	1,61803714	3,7135E-05	
17	987	1,61803279	3,2787E-05	

Exercice 2.2 : Algorithme-2

Transformez la boucle « pour » (entre les lignes 9 et 11) en une boucle « tant que ».

Exercice 2.3 : Implantation

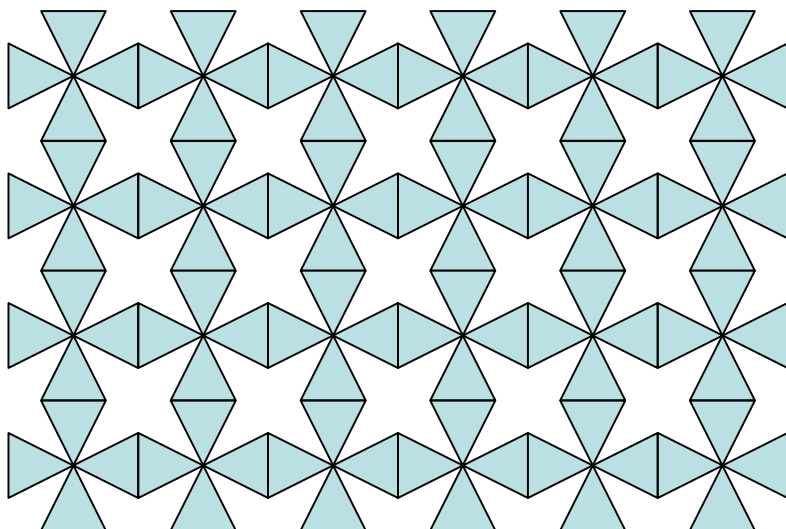
Programmez en VBA l'action Fibol0

Exercice 2.4 : Paramétrage

Transformez l'action Fibol0 en une action paramétrée (le nombre d'éléments sur lequel on fait les calculs). Introduire une Procédure Fibol, dans laquelle vous appelez l'action paramétrée pour 50 éléments.

3 Pavage de Sabliers (Examen 2^{ième} session 2004-05)

Le but de cet exercice est de produire un algorithme et un programme VBA permettant de reproduire un pavage comme donné ci-après.

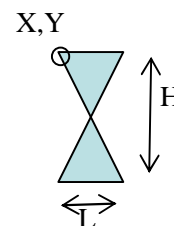


Le pavage est réalisé à partir d’une figure de base composée de 2 sabliers. Les sabliers sont de même taille et de largeur correspondant à la moitié de sa hauteur. Le second sablier a subi une rotation de 90°. Ci-dessous, on donne un algorithme non paramétré permettant de réaliser ce dessin.

```

Action DoubleSabliers
1. Début
2. Sablier(100, 100, 50, 100)
3. Sablier(100, 100, 50, 100)
4. Rotation(90)
5. Fin
    
```

Sablier(X,Y,L,H)
 Dessine un sablier
 de coordonnées (X,Y)
 de largeur L et
 de hauteur H



Exercice 3.1 : Paramétrer l’action DoubleSabliers

Proposez une version paramétrée de cette action : on attend 3 paramètres, parmi lesquels se trouvent les coordonnées du point supérieur gauche du sablier vertical

Exercice 3.2 : Algorithme pavage

Proposez un algorithme Pavage non paramétré permettant de reproduire le dessin du pavage.

On utilisera l’action DoubleSabliers() paramétrée.

On pourra passer par une action intermédiaire (éventuellement paramétrée) permettant de dessiner une ligne de DoubleSabliers.

Exercice 3.3 : Programme VBA

Réalisez les programmes VBA DoubleSabliers et Pavage.

4 ADN sous Word

Ouvrez le fichier *Sequence.doc* (présent dans l'archive TP04.zip).

Dans ce fichier, le texte représente une suite de caractère A, T, G, C qui représente une suite de bases sur un brin d'ADN. Le fichier contient aussi une macro VBA, qui automatise la mise en place de tabulations sur une ligne et permet de présenter la suite de caractères de façon « plus jolie ».

Exercice 4.1 : Comprendre la macro

- Placez le curseur au début de la première ligne des bases de l'ADN. Exécutez la macro. Que se passe-t-il ?

Exercice 4.2 : Modifier la macro

- Observez le code de la macro. Repérez sa structure : en particulier, repérez les parties répétitives du code de la macro. (code donné ci-après).
- Introduisez dans la macro 2 itérations (non imbriquées) permettant de simplifier le code en éliminant les parties répétitives.
- Vérifiez que le résultat est bien celui attendu, soit la mise en place de tabulations sur une ligne et une visualisation plus agréable de la séquence.

Exercice 4.3 : Traiter 66 lignes

Modifiez cette macro en introduisant une troisième itération afin que la mise en place des tabulations puisse s'appliquer automatiquement aux 66 lignes de la séquence.

Sub RemplacerBlancParTabulation()

' Cette macro permet de remplacer les espaces par des espaces tabulation sur une ligne de format :

' TTCTTCGACCTCCCC (15 caractères séparés par 14 espaces)

' Cette macro nécessite que votre curseur texte soit placé en début de ligne à traiter avant de lancer son exécution

' les lignes qui suivent correspondent à le mise en place de 14 taquets de tabulation sur la règle, espacés de 1cm

```
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(1), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(2), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(3), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(4), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(5), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(6), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(7), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(8), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(9), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(10), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(11), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(12), _
  Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(13), _
```

```

Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces
Selection.ParagraphFormat.TabStops.Add Position:=CentimetersToPoints(14), _
Alignment:=wdAlignTabLeft, Leader:=wdTabLeaderSpaces

' les lignes qui suivent correspondent à la mise en place de 14 espaces tabulation sur la règle
' un espace tabulation étant mis en place après avoir supprimé un espace entre 2 caractères
Selection.MoveRight Unit:=wdCharacter, Count:=2           'se déplacer après le 1er espace
Selection.TypeBackspace                                     'suppression de l'espace
Selection.TypeText Text:=vbTab                             'mise en place d'un espace tabulation
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=2
Selection.TypeBackspace
Selection.TypeText Text:=vbTab
Selection.MoveRight Unit:=wdCharacter, Count:=28 'retour en début de ligne
Selection.MoveDown Unit:=wdLine, Count:=1 'position en début de ligne suivante
End Sub

```

5 Autres Exercices

Reprenez les TP 2 et 3 pour terminer les exercices proposés et restés inachevés.