Atelier 08 : Les Algorithmes de tri et de recherche

Exercice 01 : Tri par échange

Rappeler le principe de tri par échange.
En se basent sue l'annexe 01 qui traduit les instructions algorithmique en Visual Basic (voir la fin de l'ouvrage), traduire l'algorithme de tri par échange vu dans le chapitre 8 en VB.

Solution :

Principe :

Le principe de l'algorithme de tri par échange consiste à prendre chaque élément du tableau et le comparer à tous ses suivants. Lorsque l'ordre n'est pas respecté les deux éléments sont permutés.

Traduction l'algorithme en VB :

Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com

```
Sub Main()
   ' M : indice supérieur maximum du tableau t
   Const M = 100
   Dim I As Integer, J As Integer, N As Integer
   ‘ echange : variable qui va servir à la permutation
   ‘ de deux éléments du tableau T
   Dim echange As Integer
   Dim t(1 To M) As Integer
   N = Val(InputBox("Donnez le nombre d'éléments N du tableau : "))
   If N > 100 then
      N = 100
   End If
   ' Remplir le tableau
   For I = 1 To N
      t(I) = Val(InputBox("Donnez T(" & I & ")"))
   Next I
   ' Trier le tableau
   For I = 1 To N - 1
      For J = I + 1 To N
         If t(I) > t(J) Then
            echange = t(I)
            t(I) = t(J)
            t(J) = echange
         End If
      Next J
   Next I
   ' Afficher le tableau trié
   For I = 1 To N
      MsgBox "t(" & I & ") = " & t(I)
   Next I
End Sub
```

Exercice 02 : Tri à Bulle

Rappeler le principe de tri à Bulle.
En se basent sue l’annexe 01, traduire l’algorithme de tri à Bulle vu

dans le chapitre 8 en VB.

Solution :

Principe :

Le principe de cet algorithme consiste à comparer successivement les éléments adjacents, et les permuter si le premier élément est supérieur au second. L'opération est répétée jusqu'à ce qu'il n'y ait plus de permutation possible.

Traduction l'algorithme en VB :

Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com

```
Sub Main()
   ' M : indice supérieur maximum du tableau T
   Const M = 100
   Dim i As Integer, N As Integer
   ' echange : variable qui va servir à la permutation
   ' de deux éléments du tableau T
   Dim echange As Integer
   Dim t(1 To M) As Integer
   ' Permut : variable logique
   ' Si la  permutation est possible alors Permut = vrai.
   Dim Permut As Boolean
   ' N : taille réelle du tableau
   N = Val(InputBox("Donnez le nombre d'éléments N du tableau"))
   ' Remplir le tableau
   For i = 1 To N
      t(i) = Val(InputBox("Donnez T(" & i & ")"))
   Next i
   ' Trier le tableau
   Do
      Permut = False
      For i = 1 To N - 1
         If t(i) > t(i + 1) Then
            'Permuter les deux éléments
            echange = t(i)
            t(i) = t(i + 1)
            t(i + 1) = echange
            Permut = True
         End If
      Next i
   Loop Until Permut = False
   ' Afficher le tableau trié
   For i = 1 To N
      MsgBox "t(" & i & ") = " & t(i)
   Next i
End Sub
```


Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com

Exercice 03 : Tri par insertion

Rappeler le principe de tri par insertion.
En se basent sue l'annexe 01, traduire l'algorithme de tri par insertion vu dans le chapitre 8 en VB.

Solution :

Principe :

Le principe du tri par insertion est d'insérer à la n-ième itération le n-ième élément à la bonne place.

Traduction l'algorithme en VB :

Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com

```
Sub Main()
   ' M : indice supérieur maximum du tableau T
   Const M = 100
   Dim i As Integer, j As Integer, N As Integer
   ' echange : variable qui va servir à la permutation
   ' de deux éléments du tableau T
   Dim echange As Integer
   Dim t(M) As Integer
   ' N : taille réelle du tableau
   N = Val(InputBox("Donnez le nombre d'éléments N du tableau"))
   ' Remplir le tableau
   For i = 1 To N
      t(i) = Val(InputBox("Donnez T(" & i & ")"))
   Next i
   ' Trier le tableau
   For i = 2 To N
      echange = t(i)
      j = i - 1
       Do While (j > = 1) And (t(j) > echange)
          t(j + 1) = t(j)
          j = j - 1
      Loop
      t(j + 1) = echange
   Next i
   ' Afficher le tableau trié
   For i = 1 To N
      MsgBox "t(" & i & ") = " & t(i)
   Next i
End Sub
```

Exercice 04 : Tri Rapide

Rappeler le principe de tri rapide.
En se basent sue l'annexe 01, traduire l'algorithme de tri rapide vu dans le chapitre 8 en VB.

Solution :

Principe :

Le tri rapide repose sur le principe suivant :

- diviser,
- régner,
- combiner.

Diviser : il faut d'abord choisir un élément dans le tableau t, nommé pivot Une fois le pivot choisi, il faut réorganiser le tableau de telle sorte que tous les éléments inférieurs ou égaux au pivot soient placés au début du tableau (sous tableau t1), et tous les éléments supérieurs soient placés à la fin du tableau (sous tableau t2). Ainsi, on peut découper le tableau de telle sorte que : pour tout élément x de t1 et pour tout y de t2, on ait x <= y.

Il est à noter que les deux sous-tableaux t1 et t2 ainsi obtenus ne sont pas forcément de la même taille. Cette dernière dépend du pivot qui aura été choisi.

Régner : on appelle la procédure récursivement pour t1 et pour t2.

Combiner : puisque tous les éléments du premier sous-tableau (t1) sont inférieurs à n'importe quel élément du second, une fois chacun des sous-tableaux trié, il suffit de les accoler pour obtenir le tableau trié. La phase de combinaison est donc implicite.

Traduction de l'algorithme en VB :

Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com

```
' Procédure Tri
Sub Tri(T, D As Integer, F As Integer)
   Dim Pos As Integer
   If D < F Then
      Pos = Partition(T, D, F)
      ' trie le sous tableau à gauche
      Tri T, D, Pos - 1
      ' trie le sous tableau à droite
      Tri T, Pos + 1, F
   End If
End Sub

' Fonction Partition
Function Partition(T, D As Integer, F As Integer) As Integer
   Dim Inf As Integer, Sup As Integer
   Dim Pivot As Integer, echange As Integer
   Inf = D + 1
   Sup = F
   Pivot = T(D)
   Do While (Inf <= Sup)
      ' parcourt le tableau de gauche à droite jusqu'à rencontrer
      ' un élément supérieur au pivot
      Do While (T(Inf) <= Pivot And Inf <= F)
         Inf = Inf + 1
      Loop
      ' parcourt le tableau de droite à gauche jusqu'à rencontrer
      ' un élément inférieur au pivot
      Do While (T(Sup) > Pivot And Sup > D)
         Sup = Sup - 1
      Loop
      ' Permuter ces deux éléments
      If Inf < Sup Then
         echange = T(Inf)
         T(Inf) = T(Sup)
         T(Sup) = echange
          Inf = Inf + 1
         Sup = Sup - 1
      End If
   Loop
   echange = T(D)
```


Prof. Omar El Kharki & Mme Jamila Mechbouh / elkharki@gmail.com

Exercice 05 : Recherche séquentielle

En se basent sue l'annexe 01, traduire l'algorithme de recherche séquentielle vu en chapitre 8 en VB.

Solution :

Traduction de l'algorithme en VB :

Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com

```
' Déclaration de la fonction RechSéq
' T : tableau, N : nombre d'éléments du tableau
' ValRech : valeur recherchée
Function RechSéq(T, N As Integer, ValRech As Integer) As Integer
   Dim i As Integer
   For i = 1 To N
      If T(i) = ValRech Then
         RechSéq = i
         Exit Function
      End If
   Next i
   ‘ La fonction RechSéq retourne -1 si la valeur
   ' recherchée n’existe pas dans le tableau.
   RechSéq = -1
End Function

'Programme principal
Sub Main()
   Dim A(1 To 100) As Integer
   Dim i As Integer
   ' m : taille réelle du tableau
   Dim m As Integer
   ' x : valeur recherchée
   Dim x As Integer
   m = Val(InputBox("Donner la taille du tableau"))
   ' Remplissage du tableau
   For i = 1 To m
      A(i) = Val(InputBox("Donner l'élément A(" & i & ")"))
   Next i
   x = Val(InputBox("Donner la valeur recherchée"))
   ' Appel de la fonction RechSéq
   MsgBox RechSéq(A, m, x)
End Sub
```

Exercice 06 : Recherche dichotomique

En se basent sue l’annexe 01, traduire l’algorithme de recherche

Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com

dichotomique vu en chapitre 8 en VB. (On suppose que le tableau est trié).

Solution :

Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com

```
' Déclaration de la fonction RechDicho
' T : tableau, N : nombre d'élément du tableau
' ValRech : valeur recherchée
' M : indice de l'élément médian
Function RechDicho(T, N As Integer, ValRech As Integer) As Integer
   Dim D As Integer, F As Integer, m As Integer
   D = 1
   F = N
   Do While D <= F
      ' int : c'est la fonction partie entière.
      m = Int((D + F) / 2)
      If T(m) = ValRech Then
         RechDicho = m
         Exit Function
      Else
         If T(m) < ValRech Then
            D = m + 1
            If T(D) = ValRech Then
               RechDicho = D
               Exit Function
            End If
         Else
            F = m - 1
            If T(F) = ValRech Then
               RechDicho = F
               Exit Function
            End If
         End If
      End If
   Loop
   ‘ La fonction RechDicho retourne -1 si la valeur
   ' recherchée n’existe pas dans le tableau.
   RechDicho = -1
End Function

'Programme principal
Sub Main()
   Dim A(100) As Integer
   Dim i As Integer
   ' N : taille réelle du tableau
```

Prof. Omar El kharki & Mme Jamila Mechbouh / elkharki@gmail.com