

Devoir en classe d'Informatique II,1

50 minutes – 30 points

Exercice 1

1. Présentez l'algorithme de la *Recherche dichotomique – version récursive*. [10 p.]

```

function dichotomique(list : TListBox; element : string; left, right : integer):integer ;
var
  mid : integer;
begin
  if (left > right) then
    result := -1
  else
    begin
      mid := (left + right) div 2;
      if list.Items[mid]=element then
        result := mid
      else if element < list.Items[mid] then
        result := recherche_dichotomique(list, element, left, mid-1)
      else
        result := recherche_dichotomique(list, element, mid+1, right)
    end;
  end;
end;

```

2. Donnez un exemple d'exécution de la recherche de l'élément "k" en utilisant une TListBox, appelée « lbList », contenant les valeurs (de type string) suivantes: [4 p.]

a	b	d	f	h	k	m	p	s	x
---	---	---	---	---	---	---	---	---	---

$dichotomique(lbList, 'k', 0, 9) = dichotomique(lbList, 'k', 5, 9) = dichotomique(lbList, 'k', 5, 6) = 5$

3. Illustrez le fonctionnement de la condition d'arrêt de cet algorithme en utilisant l'exemple d'exécution de la recherche de l'élément "j" dans cette même TListBox. [4 p.]

La fonction « dichotomique » continue aussi longtemps que « left » est inférieur ou égal à « right ».

On a donc :

$dichotomique(lbList, 'j', 0, 9) = dichotomique(lbList, 'j', 5, 9) = dichotomique(lbList, 'j', 5, 6) =$

$dichotomique(lbList, 'j', 5, 4) = -1$

Dans le dernier appel, le paramètre d'appel « left » est supérieur à « right » et la fonction retourne -1.

Exercice 2

[12 p.]

(Examen de fin d'études secondaires 2007 – septembre)

1. Ecrivez une fonction **horner** qui prend comme entrée un polynôme et un nombre de type real qui évalue le polynôme en ce réel. [5 p.]

```

function horner(a : poly; x : real) : real;
var
  i : integer;
begin
  result := a.c[a.d];
  for i := a.d downto 1 do
    result := a.c[i-1] + x * result;
  end;

```

2. Précisez comment le polynôme intervenant dans cette fonction est implémenté. [2 p.]

Le type POLY représente des polynômes à coefficients réels et de degré inférieur ou égal à 100.

```

type POLY = record
  C : array [0 .. 100] of real;
  D : integer;
end ;

```

3. Adaptez la fonction **horner** pour qu'elle serve à transformer un string (non vide) composé de chiffres en un entier (la fonction prédéfinie *ord* est permise mais pas les fonctions de conversion). [5 p.]

Idée : '253' donne 253 par le calcul $2 \times 10^2 + 5 \times 10^1 + 3 \times 10^0$ (considérer 2, 5 et 3 comme les coefficients d'un polynôme et évaluer en le réel 10).

```

function horner(s : string) : integer;
var
  i : integer;
begin
  result := ord(s[1]) - ord('0');
  for i := 2 to length(s) do
    result := (ord(s[i]) - ord('0')) + 10 * result;
  end;

```