

## Question 1

- (1) L'idée de l'algorithme de tri rapide est de diviser la liste en 2 parties grâce à la fonction *DIVISION* : celle-ci compare le dernier élément de la liste, d'indice  $D$ , appelé *CANDIDAT* aux autres éléments de la liste, plus précisément ceux d'indices  $G$  à  $D - 1$ . Par échanges successifs, les éléments  $\leq$  *CANDIDAT* sont placés à gauche de celui-ci, et les éléments  $\geq$  *CANDIDAT* sont placés à droite du *CANDIDAT*. De ce fait, le *CANDIDAT* se trouvera à sa position définitive dans la liste triée. L'indice de cette position est retournée par la fonction *DIVISION*. Les sous-listes constituées par les éléments précédant resp. suivant le *CANDIDAT* seront triées par le même algorithme, récursivement.

(2)

```
1  procedure ECHANGE(var LISTE:TListBox; I,J:integer);
2  var AUX:string;
3  begin
4    AUX:=LISTE.Items[I];
5    LISTE.Items[I]:=LISTE.Items[J];
6    LISTE.Items[J]:=AUX
7  end;

1  function DIVISION(var LISTE:TListBox; G,D:integer):integer;
2  var I,J:integer;
3      CANDIDAT:string;
4  begin
5    CANDIDAT:=LISTE.Items[D];
6    J:=D-1;
7    I:=G;
8    while I<=J do
9      if LISTE.Items[I]<CANDIDAT then
10         I:=I+1
11      else if LISTE.Items[J]>CANDIDAT then
12         J:=J-1
13      else begin
14         ECHANGE(LISTE,I,J);
15         I:=I+1;
16         J:=J-1
17      end;
18    ECHANGE(LISTE,I,D);
19    DIVISION:=I
20  end;
```

```

1  procedure TRI_RAPIDE_R(var LISTE:TListBox; G,D:integer);
2  var I:integer;
3  begin
4    if G<D then begin
5      I:=DIVISION(LISTE,G,D);
6      TRI_RAPIDE_R(LISTE,G,I-1);
7      TRI_RAPIDE_R(LISTE,I+1,D)
8    end
9  end;

```

- (3) ('B','E','A','D','C')  
('B','A','E','D','C')  
('B','A','C','D','E')  
('A','B','C','D','E')  
('A','B','C','D','E')

## Question 2

40 points

The screenshot shows a window titled "Intégrale définie d'un polynôme ou d'une fraction rationnelle". The interface is divided into two main sections:

- Left Section: Fraction rationnelle  $p(x)/q(x)$ :**
  - Labels for  $p(x)$  and  $q(x)$  are positioned above two small input boxes.
  - Below these are two large vertical text input fields for entering coefficients.
  - Text: "Entrez les coefficients ci-dessus suivant les puissances croissantes et validez par Enter !"
  - At the bottom are two "Effacer" (Erase) buttons.
- Right Section: Bornes et nombre de subdivisions:**
  - Input fields for  $a$ ,  $b$ , and  $n$ .
  - The  $n$  field contains the value "1000".
- Bottom Section: Intégrale de  $a$  à  $b$  de la fraction rationnelle  $p(x)/q(x)$ :**
  - A button labeled "Calcul par la méthode des trapèzes :".
  - A label "Résultat :" followed by a blank space for the output.

8 pts

```

procedure TForm1.edtCPEnter(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if key=VK_Return then begin
    lbP.Items.Add(edtCP.Text);
    edtCP.Text:='';
    edtCP.SetFocus
  end

end;

```

```

procedure TForm1.edtCQEnter(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if key=VK_Return then begin
    lbQ.Items.Add(edtCQ.Text);
    edtCQ.Text:='';
    edtCQ.SetFocus
  end

end;

```

4 pts

```

procedure TForm1.btnEffacerPClick(Sender: TObject);
begin
  lbP.Clear;
end;

```

```

procedure TForm1.btnEffacerQClick(Sender: TObject);
begin
  lbQ.Clear;
end;

```

2 pts

```

function Horner(lbP:TListBox;x:real):real;
var dP,i:integer;
begin
  dP:=lbP.Items.Count-1;
  result:=0;
  for i:=dP downto 0 do result:=result*x+StrToFloat(lbP.Items[i]);
end;

```

5 pts

```

function Frac(x:real;lbP,lbQ:TListBox):real;
begin
  if Horner(lbQ,x)<>0 then result:=Horner(lbP,x)/Horner(lbQ,x)
  else result:=0
end;

```

3 pts

```

function TestRacine(lbQ:TListBox;a,b,epsilon:real):boolean;
//Cette fonction retourne TRUE si le polynôme Q a une racine entre a et b
//et FALSE sinon. Elle risque de détecter des racines de multiplicité paire "fausses"
//lorsque le graphe de Q se rapproche trop de 0 (sans toutefois le toucher).
//Par ailleurs, si epsilon est trop grand, alors elle risque aussi
// de ne pas trouver certaines racines de multiplicité impaire (trop rapprochées).
begin
  if b-a<epsilon then result:=(Horner(lbQ,a)*Horner(lbQ,b)<=0) or
    ((Horner(lbQ,a)*Horner(lbQ,b)>0) and (Horner(lbQ,a)*Horner(lbQ,b)<sqr(epsilon)))
  else result:=(TestRacine(lbQ,a,(a+b)/2,epsilon)) or (TestRacine(lbQ,(a+b)/2,b,epsilon))
end;

```

8 pts

```

procedure TForm1.btnCalculClick(Sender: TObject);
//Calcul de l'intégrale de la fraction rationnelle par la méthode des trapèzes
var a,b,h,res:real;
    i,n:integer;
begin
  a:=StrToFloat (edtA.Text) ;
  b:=StrToFloat (edtB.Text) ;
  n:=StrToInt (edtN.Text) ;
  if TestRacine (lbQ,a,b,10E-3) then showMessage('Le dénominateur s''annule sur [a,b] !')
  else begin
    h:=(b-a)/n;
    res:=(Frac (a, lbP, lbQ)+Frac (b, lbP, lbQ) )/2;
    for i:=1 to n-1 do res:=res+Frac (a+i*h, lbP, lbQ) ;
    res:=h*res;
    lblResultat.Caption:='Résultat : ' + FloatToStr (res)
  end
end;

```

10 pts
--------