

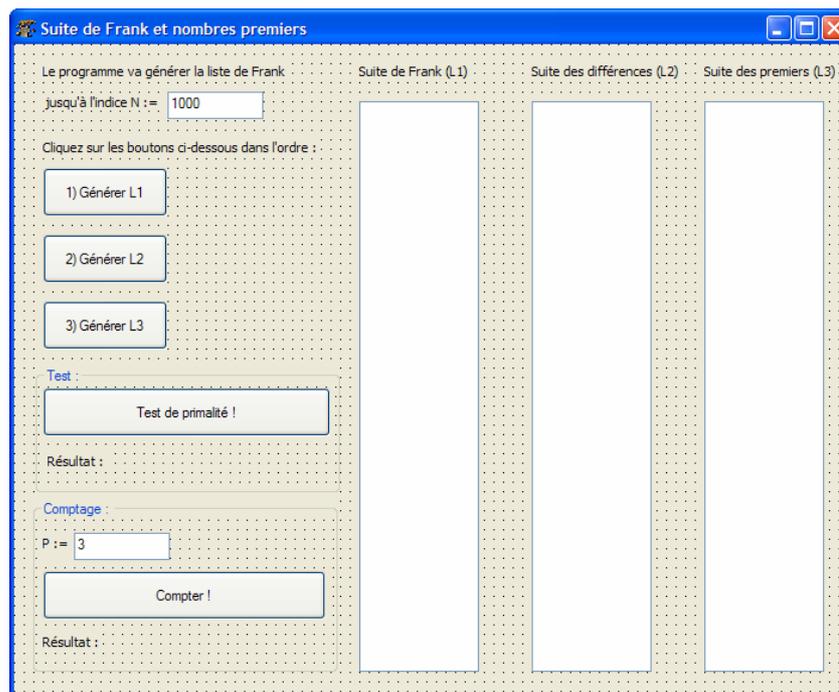
Partie écrite

Question 2

- (1) $p(6) = np(5) = p(4) = np(3) = p(2) = np(1) = p(0) = \text{true}$
- (2) $p(5) = np(4) = p(3) = np(2) = p(1) = np(0) = \text{false}$
- (3) La fonction **p** étudie la parité d'un entier naturel : si le nombre donné est pair, la fonction retourne true, sinon elle retourne false.
- (4) On n'a pas le droit de définir la fonction **np** *avant* ou *après* la fonction **p**, puisque la fonction **p** appelle la fonction **np** et la fonction **np** appelle la fonction **p**.

Partie pratique (70 min)

- (1) *Formulaire Delphi :*



- (2) a)

```

function premier(n:integer):boolean;
var i:integer;
begin
  if n<=2 then result:=n=2
  else if n mod 2=0 then result:=false
  else begin
    i:=3;
    result:=true;
    while (i*i<=n) and result do
      if n mod i=0 then result:=false else i:=i+2
    end
  end
end;

```

b)

```
function pgcd(a,b:int64):int64;
begin
  if b=0 then result:=a else result:=pgcd(b,a mod b)
end;
```

c)

```
function f(n:integer):int64;
var temp:int64;
begin
  if n=1 then result:=7
  else begin
    temp:=f(n-1); // pour augmenter la performance de la fonction
    result:=temp+pgcd(n,temp)
  end
end;
```

Algorithmes correspondant aux boutons :

```
procedure TForm1.btnFrankClick(Sender: TObject);
var i,N:integer;
begin
  lbFrank.Clear;
  N:=strtoint(edtN.Text);
  for i:=1 to N do lbFrank.Items.Add(inttostr(f(i)))
end;
```

```
procedure TForm1.btnDiffClick(Sender: TObject);
var i,N:integer;
begin
  lbDiff.Clear;
  N:=strtoint(edtN.Text);
  for i:=1 to N-1 do lbDiff.Items.Add(inttostr(strtoint(lbFrank.Items[i])-strtoint(lbFrank.Items[i-1])))
end;
```

```
procedure TForm1.btnPremClick(Sender: TObject);
var i,N:integer;
begin
  lbPrem.Clear;
  N:=lbDiff.Items.Count-1;
  for i:=0 to N do if strtoint(lbDiff.Items[i])<>1 then lbPrem.Items.Add(lbDiff.Items[i])
end;
```

```
procedure TForm1.btnTestClick(Sender: TObject);
var test:boolean;
    i:integer;
begin
  test:=true;
  i:=0;
  while test and (i<lbPrem.Count) do
  begin
    test:=premier(strtoint(lbPrem.Items[i]));
    i:=i+1
  end;
  if test then lblTest.Caption:='Résultat : Tous les éléments de L3 sont premiers.'
  else lblTest.Caption:='Résultat : L''élément '+lbPrem.Items[i-1]+' d''indice '+inttostr(i-1)+
    ' n''est pas premier.'
end;
```

```
procedure TForm1.btnCountClick(Sender: TObject);
var i,res:integer;
    P:string;
begin
  P:=edtP.text;
  res:=0;
  for i:=0 to lbPrem.Items.Count-1 do if lbPrem.Items[i]=P then res:=res+1;
  lblCount.Caption:= 'Résultat : '+P+' se trouve '+ inttostr(res) + ' fois dans L3.'
end;
```

Exemple d'exécution du programme avec $N=100$:

Suite de Frank et nombres premiers

Le programme va générer la liste de Frank
 jusqu'à l'indice N :=

Cliquez sur les boutons ci-dessous dans l'ordre :

1) Générer L1

2) Générer L2

3) Générer L3

Test :

Test de primalité !

Résultat : Tous les éléments de L3 sont premiers.

Comptage :

P :=

Compter !

Résultat : 3 se trouve 5 fois dans L3.

Suite de Frank (L1)	Suite des différences (L2)	Suite des premiers (L3)
7	1	5
8	1	3
9	1	11
10	5	3
15	3	23
18	1	3
19	1	47
20	1	3
21	1	5
22	11	3
33	3	
36	1	
37	1	
38	1	
39	1	
40	1	
41	1	
42	1	
43	1	
44	1	
45	1	
46	23	
69	3	
72	1	
73	1	
74	1	
75	1	
76	1	
77	1	
78	1	
79	1	
80	1	
81	1	
82	1	
83	1	
84	1	
95	1	

G. Lorang