

Question 1

- (1) Version récursive de la fonction
- pgcd**
- :

```

function pgcd(a,b:integer):integer;
//Calcule le pgcd de deux entiers naturels
//Donne le résultat faux : pgcd(0,0)=0
begin
  if b=0 then result:=a
  else result:=pgcd(b,a mod b)
end;

```

- (2)
- $$\begin{aligned}
 \text{pgcd}(8,13) &= 8 = 13 \cdot 0 + 8 \\
 &= \text{pgcd}(13,8) & 13 &= 8 \cdot 1 + 5 \\
 &= \text{pgcd}(8,5) & 8 &= 5 \cdot 1 + 3 \\
 &= \text{pgcd}(5,3) & 5 &= 3 \cdot 1 + 2 \\
 &= \text{pgcd}(3,2) & 3 &= 2 \cdot 1 + 1 \\
 &= \text{pgcd}(2,1) & 2 &= 1 \cdot 2 + 0 \\
 &= \text{pgcd}(1,0) \\
 &= 1
 \end{aligned}$$

- (2)
- ```

function relprimes(a,b:integer):boolean;
begin
 result:=pgcd(a,b)=1
end;

```

## Question 2

- (1) a)

$$\begin{aligned}
 s(3594287) &= 7 + s(359428) \\
 &= 7 + 8 + s(35942) \\
 &= 7 + 8 + 2 + s(3594) \\
 &= 7 + 8 + 2 + 4 + s(359) \\
 &= 7 + 8 + 2 + 4 + 9 + s(35) \\
 &= 7 + 8 + 2 + 4 + 9 + 5 + s(3) \\
 &= 7 + 8 + 2 + 4 + 9 + 5 + 3 \\
 &= 38
 \end{aligned}$$

b)  $s$  calcule la somme des chiffres d'un entier  $x$  donné.

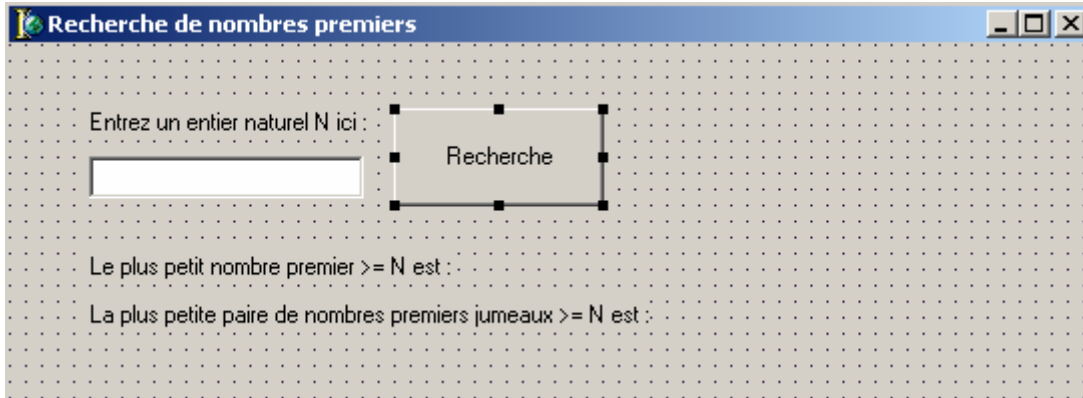
- (2) a)
- $m(3594287) = m(38) = m(11) = m(2) = 2$
- .

b)  $m$  calcule  $s(s(s(\dots s(x))))$ , jusqu'à ce que le résultat soit un entier  $< 10$ .

c) Lorsque  $\mathbf{m(x)=9}$ ,  $x$  est un multiple non nul de 9. C'est un critère de divisibilité bien connu !

## Partie pratique (60 min)

8 p.



8 p.

```
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;
```

6 p.

```
function premierpg(n:integer):integer;
begin
if n<=2 then result:=2
else begin
if n mod 2=0 then n:=n+1;
while not (premier(n)) do n:=n+2;
result:=n
end;
```

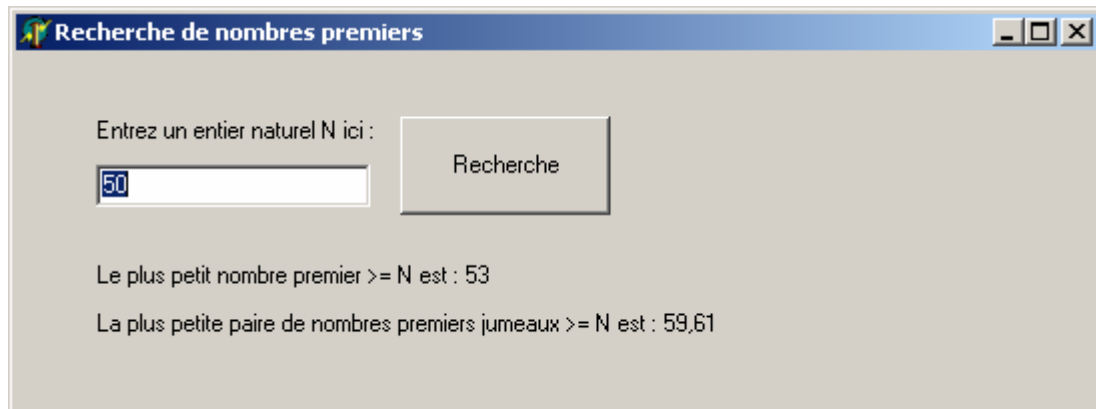
8 p.

```
procedure pairepg(n:integer;var p,q:integer);
//Cette procédure sauvegarde dans les variables p et q la plus petite paire de
//nombres premiers jumeaux >=n);
begin
p:=premierpg(n);
q:=premierpg(p+1);
while q-p<>2 do
begin
p:=q;
q:=premierpg(p+1);
end;
```

5 p.

```
procedure TfrmRecherche.btnRechercheClick(Sender: TObject);
var n,p,q,r:integer;
begin
 n:=strtoint(edtN.text);
 p:=premierpg(n);
 pairepg(p,q,r);
 lblResult1.Caption:='Le plus petit nombre premier >= N est : '+ inttostr(p);
 lblResult2.Caption:='La plus petite paire de nombres premiers jumeaux >= N est : '
 + inttostr(q)+' '+ inttostr(r);
 edtN.SetFocus
end;
```

Exemple d'application :



G. Lorang