

Bac pratique informatique sections scientifiques  
26 mai 2014 – 8h

---

```
program bac2014 ;  
uses wincrt ;
```

```
var m,n : integer ;
```

```
procedure saisir(var n,m : integer ) ;
```

```
begin  
repeat  
write('Donner la valeur de N : ');  
readln(n);  
until n>=100 ;  
repeat  
write('Donner la valeur de M : ');  
readln(m);  
until m>n ;  
end ;
```

```
function harshad ( Nb : integer ) : boolean ;  
var j,somme,a,e : integer ;  
ch : string ;  
test : boolean ;
```

```
begin  
str(NB,ch) ;  
somme:=0 ;  
for j:=1 to length(ch) do  
begin  
val(ch[j],a,e) ;  
somme:=somme+a ;  
end ;  
if nb mod somme = 0 then  
test:=true  
else  
test := false ;  
harshad:=test ;  
end ;
```

```
function premier( NB : integer ) : boolean ;  
var k:integer ;  
test : boolean ;
```

```
begin  
test := true ;  
k:=nb-1 ;  
repeat  
if nb mod k = 0 then  
test := false ;  
k:=k-1 ;  
until (k=2) or (test=false) ;  
  
premier:=test ;  
end ;
```

```
procedure affiche(n,m : integer ) ;  
var i:integer ;
```

```
begin  
for i:=n to m do  
begin  
if (harshad(i)=true) and (premier(i-1)=true) then  
writeln(i,' est Harshad et son predesseur ',i-1,' est un nombre premier. ' ) ;  
end ;  
end ;
```

```
(***** PROGRAMME PRINCIPALE *****)
```

```
BEGIN  
SAISIR(N,M) ;  
AFFICHE(N,M) ;  
END .
```

Bac pratique informatique sections scientifiques  
26 mai 2014 – 9h30

---

```
program bac2014 ;  
uses wincrt ;
```

```
var nb : integer ;
```

```
procedure saisie ( var Nb : integer ) ;
```

```
begin  
repeat  
write('Saisir la valeur du nombre entier : ');  
readln(nb) ;  
until nb >= 4 ;  
end ;
```

```
function premier ( n : integer ) : boolean ;
```

```
var test : boolean ;
```

```
    k:integer ;
```

```
begin
```

```
test := true ;
```

```
if n=1 then
```

```
test:=false
```

```
else
```

```
    for k:=2 to n-1 do
```

```
        begin
```

```
            if n mod k =0 then
```

```
                test :=false ;
```

```
            end ;
```

```
premier := test ;
```

```
end ;
```

```
procedure couples( nb : integer ) ;
var T1,T2: array[0..100] of integer ;
    i,c:integer ;

begin

c:=0 ;
for i:=1 to nb-1 do
    begin
    if (premier(i)=true) and (premier(nb-i)=true) then
        begin
        c:=c+1 ;
        T1[c]:=i ;
        T2[c]:=nb-i ;
        end ;
        end ;
write(nb,' possede ',c,' couples des nombres entier : ' ) ;
for i:=1 to c do
    begin
    write('(' ,T1[i],',',T2[i],') , ' ) ;
    end ;
end ;
```

(\*\*\*\*\* PROGRAMME PRINCIPALE \*\*\*\*\*)

```
BEGIN
SAISIE(Nb) ;
COUPLES(Nb) ;
END.
```

Bac pratique informatique sections scientifiques  
26 mai 2014 – 11h

---

```
program bac2014 ;
uses wincrt ;
type tab=array[0..100] of integer ;

var T:tab ;
n : integer ;

procedure saisie(var n : integer ) ;

begin
repeat
write('Donner la taille du tableau : ');
readln(n) ;
until n in [5..30] ;
end ;

function verif(T:tab ; n:integer ; NB : integer ):boolean ;
var test:boolean ;
j: integer ;
begin
test:= true ;
for j:=1 to n do
begin
if NB=T[j] then
test := false
end ;
verif:= test ;
end ;
```

```
procedure remplir(var T:tab ; n:integer ) ;
var i:integer ;

begin
repeat
write('Donner le premier element du tableau : ') ;
readln(T[1]) ;
until T[1] in [1..99] ;
for i:=2 to n do
begin
repeat
write('Donner le ',i,' eme element du tableau : ') ;
readln(T[i]) ;
until (T[i] in [1..99]) and (verif(T,i-1,T[i])=true) ;
end ;
end;
```

```
procedure tri( var T:tab ; n: integer ) ;
var TI:tab ;
i,c:integer ;

begin

for i:=1 to 99 do
begin
TI[i]:=0 ;
end ;

for i:=1 to n do
begin
TI[T[i]]:=T[i] ;
end ;

c:=1 ;
for i:=1 to 99 do
begin
if TI[i]<>0 then
begin
T[c]:=TI[i] ;
c:=c+1 ;
end ;
end;
```

```
end ;
```

```
procedure affiche(T: tab ; n:integer ) ;
var i:integer ;

begin
writeln('Le tableau trié : ');
for i:=1 to n do
begin
write(T[i], ' ');
end ;
end ;
```

```
(***** PROGRAMME PRINCIPALE *****)
BEGIN
SAISIE(N) ;
REPLIR(T,N) ;
TRI(T,N) ;
AFFICHE(T,N) ;
END.
```

Bac pratique informatique sections scientifiques  
26 mai 2014 – 14h

---

```
program bac2014 ;
uses wincrt ;
type tab=array[0..100] of integer ;

var i,n,c1,c2: integer ;
    T,TA,TB: tab ;

procedure saisie(var n: integer ) ;

begin
repeat
write('Donner la taille du tableau : ');
readln(n) ;
until n in [5..25] ;
end ;

procedure remplir(var T:tab ; n:integer ) ;
var i:integer ;
begin
repeat
write('Donner le premier element du tableau : ');
readln(T[1]) ;
until T[1]>0 ;
for i:=2 to n do
begin
repeat
write('Donner le ',i,' eme element du tableau : ');
readln(T[i]) ;
until T[i]>0 ;
end ;
end ;

function abundant(nb : integer ) : boolean ;
var test : boolean ;
    s,i:integer ;
begin
s:=0 ;
for i:=1 to nb-1 do
begin
if nb mod i=0 then
```



```
s:=s+i ;
end ;
if nb>s then
test := true else test:= false ;
abondant:= test ;
end ;
```

```
function deficient(nb : integer ) : boolean ;
var test : boolean ;
    s,i:integer ;
begin
s:=0 ;
for i:=1 to nb-1 do
begin
if nb mod i=0 then
s:=s+i ;
end ;
if nb<s then
test := true else test:=false ;
deficient:= test ;
end ;
```

```
procedure determination( T:tab; n:integer ; var TA,TB: tab ; var c1,c2 : integer ) ;
var j: integer ;
```

```
begin
c1:=0 ;
c2:=0 ;
for j:=1 to n do
begin
if abondant(T[j])= true then
begin
c1:=c1+1 ;
TA[c1]:=T[j];
end else
if deficient(T[j])=true then
begin
c2:=c2+1 ;
TB[c2]:=T[j];
end ;
end ;
end ;
```

```
procedure affiche(TA,TB:tab ;c1,c2 : integer ) ;
var i : integer ;

begin
writeln('Le tableau des elements abondants : ');
for i:=1 to c1 do
begin
write(TA[i], ' ');
end ;
writeln("");
writeln('Le tableau des elements déficients : ');
for i:=1 to c2 do
begin
write(TB[i], ' ');
end ;
end ;
```

(\*\*\*\*\* PROGRAMME PRINCIPALE \*\*\*\*\*)

```
BEGIN
SAISIE(N) ;
REPLIR(T,N) ;
DETERMINATION(T,N,TA,TB,C1,C2) ;
AFFICHE(TA,TB,C1,C2) ;
END.
```