

TURBO PASCAL

Correction Des Exercices

Prof : FENNI Salah

Lycée Chebba
©R 2006

LES STRUCTURES SIMPLES

Exercice 1-d)

- [V] Lire (A)
- [F] Lire ("A")
- [V] Ecrire ("A = ", A)
- [V] Ecrire (5 mod 7 div 2)

- [F] Lire (45)
- [F] Lire ("A = ", A)
- [V] Ecrire (A, " ", B)
- [V] Ecrire ("Saisir un réel")

- [F] Lire (A+B)
- [V] Ecrire (X+2*Y)
- [V] Ecrire (A : 6 : 2)
- [V] Ecrire (45)

<p style="text-align: center;">Exercice 2</p> <p>0) Début Sortie_Inverse</p> <p>1) Ecrire ("A = "), Lire (A)</p> <p>2) Ecrire ("B = "), Lire (B)</p> <p>3) Ecrire ("C = "), Lire(C)</p> <p>4) Ecrire (C, " ", B, " ", A)</p> <p>5) Fin Sortie_Inverse</p>	<p style="text-align: center;">Exercice 3</p> <p>0) Début Cylindre</p> <p>1) Ecrire ("Donner le rayon : "), Lire(R)</p> <p>2) Ecrire ("Donner la hauteur : "), Lire (H)</p> <p>2) $V \leftarrow \text{PI} * R * R * H$</p> <p>3) Ecrire ("Volume = ", V)</p> <p>4) Fin Cylindre</p>
<p style="text-align: center;">Exercice 4</p> <p>0) Début Surface_rectangle</p> <p>1) Ecrire ("Largeur = "), Lire (la)</p> <p>2) Ecrire ("Longueur = "), Lire (lo)</p> <p>3) $S \leftarrow \text{la} * \text{lo}$</p> <p>4) Ecrire ("La surface du rectangle dont la longueur mesure ", lo, " m et la largeur mesure ", la, " m, a une surface égale à ", s, " mètres carrés.")</p> <p>5) Fin Surface_rectangle</p>	<p style="text-align: center;">Exercice 5</p> <p>0) Début Piscine</p> <p>1) Ecrire ("Donner les dimensions de la piscine"), Lire (LO, LA, PR)</p> <p>2) $V \leftarrow \text{LO} * \text{LA} * \text{PR}$</p> <p>3) $\text{EAU} \leftarrow V * 1000$</p> <p>4) Ecrire ("Le volume = ", V)</p> <p>5) Ecrire ("Quantité d'eau = ", EAU, " litres")</p> <p>6) Fin Piscine</p>
<p style="text-align: center;">Exercice 6</p> <p>0) Début Aire_Trapeze</p> <p>1) Ecrire ("Donner les dimensions du trapèze "), Lire (H, B1, B2)</p> <p>2) $S \leftarrow H * (B1 + B2)$</p> <p>3) Ecrire ("La surface = ", S)</p> <p>4) Fin Aire_trapeze</p>	<p style="text-align: center;">Exercice 7</p> <p>0) Début Permut</p> <p>1) Lire (A, B)</p> <p>2) $\text{AUX} \leftarrow A$</p> <p>3) $A \leftarrow B$</p> <p>4) $B \leftarrow \text{AUX}$</p> <p>5) Ecrire ("La nouvelle valeur de A est : ", A)</p> <p>6) Ecrire ("La nouvelle valeur de B est : ", B)</p> <p>7) Fin Permut</p>
<p style="text-align: center;">Exercice 8</p> <p>0) Début Permut_Circulaire</p> <p>1) Lire (A, B, C)</p> <p>2) $\text{AUX} \leftarrow A$</p> <p>3) $A \leftarrow C$</p> <p>4) $C \leftarrow B$</p> <p>5) $B \leftarrow \text{AUX}$</p> <p>6) Ecrire (A, " ", B, " ", C)</p> <p>7) Fin Permut_Circulaire</p>	<p style="text-align: center;">Exercice 9</p> <p>0) Début Permut</p> <p>1) Lire (X, Y)</p> <p>2) $X \leftarrow X + Y$</p> <p>3) $Y \leftarrow X - Y$</p> <p>4) $X \leftarrow X - Y$</p> <p>5) Ecrire (X, " ", Y)</p> <p>6) Fin Permut</p>
<p style="text-align: center;">Exercice 10</p> <p>0) Début Division</p> <p>1) Ecrire ("A = "), Lire (A)</p> <p>2) Ecrire ("B = "), Lire (B)</p> <p>3) $Q \leftarrow A \text{ div } B$</p> <p>4) $R \leftarrow A \text{ mod } B$</p> <p>5) Ecrire ("Le quotient est ", q, " et le reste est ", r)</p> <p>6) Fin Division</p>	<p style="text-align: center;">Exercice 11</p> <p>0) Début Temperature</p> <p>1) Ecrire ("Donner une température en °C : "), Lire (D)</p> <p>2) $F \leftarrow 9/5 * D + 32$</p> <p>3) Ecrire (D, " °C = ", F, " Fahrenheit")</p> <p>4) Fin Temperature</p>
<p style="text-align: center;">Exercice 12</p> <p>0) Début Mile_marin</p> <p>1) Ecrire ("Donner le nombre de Km : "), Lire (km)</p> <p>2) $\text{Mm} \leftarrow \text{km} / 1.852$</p> <p>3) Ecrire (km, " km = ", mm, " miles marins")</p> <p>4) Fin mile_marin</p>	<p style="text-align: center;">Exercice 13</p> <p>0) Début Conversion</p> <p>1) Ecrire ("Nombres de bits = "), Lire (bit)</p> <p>2) Octet $\leftarrow \text{bit} / 8$</p> <p>3) Kilo $\leftarrow \text{octet} / 1024$</p> <p>4) Mega $\leftarrow \text{kilo} / 1024$</p> <p>5) Giga $\leftarrow \text{mega} / 1024$</p> <p>6) Ecrire (octet, kilo, mega, giga)</p> <p>7) Fin conversion</p>
<p style="text-align: center;">Exercice 14</p> <p>0) Début Temps</p> <p>1) Ecrire ("Donner une durée en secondes "), Lire (T)</p> <p>2) $H \leftarrow T \text{ div } 3600$</p> <p>3) $M \leftarrow (T \text{ mod } 3600) \text{ div } 60$</p> <p>4) $S \leftarrow T \text{ mod } 60$</p> <p>5) Ecrire (H, " : ", M, " : ", S)</p> <p>6) Fin Temps</p>	<p style="text-align: center;">Exercice 16</p> <p>0) Début Interet_Simple</p> <p>1) Ecrire ("Donner la somme initiale : "), Lire (SOM)</p> <p>2) Ecrire ("Donner le taux d'intérêt : "), Lire (TAUX)</p> <p>3) $\text{INTERET} \leftarrow (\text{SOM} * \text{TAUX} / 100) * 5$</p> <p>4) $\text{VA} \leftarrow \text{SOM} + \text{INTERET}$</p> <p>5) Ecrire ("Après 5 ans la somme sera = ", VA)</p> <p>6) Fin Interet_Simple</p>
<p style="text-align: center;">Exercice 17</p> <p>0) Début Consommation</p> <p>1) Ecrire ("Nombre de Km parcouru : "), Lire(Y)</p> <p>2) Ecrire ("Nombre de litres d'essence : "), Lire (L)</p> <p>3) $\text{TAUX} \leftarrow (L * 100) / Y$</p>	<p style="text-align: center;">Exercice 18</p> <p>0) Début Résistance</p> <p>1) Ecrire ("Donner les trois résistances : "), Lire (R1, R2, R3)</p> <p>2) $R \leftarrow 1 / (1/R1 + 1/R2 + 1/R3)$</p> <p>3) Ecrire ("La résistance équivalente est = ", R)</p>

4) Ecrire ("Taux de consommation est = ", TAUX, " %") 5) Fin Consommation	4) Fin Résistance
<p style="text-align: center;">Exercice 15</p> 0) Début Futur 1) Ecrire ("Donner un verbe du 1er groupe : "), Lire (verbe) 2) Ecrire ("Je ", verbe, "ai") 3) Ecrire ("Tu ", verbe, "as") 4) Ecrire ("Il ou elle ", verbe, "a") 5) Ecrire ("Nous ", verbe, "ons") 6) Ecrire ("Vous ", verbe, "ez") 7) Ecrire ("Ils ou elles ", verbe, "ont") 8) Fin Futur	<p style="text-align: center;">Exercice 19</p> 0) Début NBR_3 1) Ecrire ("Saisir un entier formé de 3 chiffres non nuls : "), Lire (N) 2) $c \leftarrow n \text{ div } 100$ 3) $d \leftarrow n \text{ mod } 100 \text{ div } 10$ 4) $u \leftarrow n \text{ mod } 10$ 5) $r1 \leftarrow c*100+u*10+d$ 6) $r2 \leftarrow u*100+d*10+c$ 7) $r3 \leftarrow u*100+c*10+d$ 8) $r4 \leftarrow d*100+c*10+u$ 9) $r5 \leftarrow d*100+u*10+c$ 10) Ecrire (N, r1, r2, r3, r4, r5) 11) Fin NBR_3
<p style="text-align: center;">Exercice 20</p> 0) Début Sup_Inf 1) Ecrire ("A = "), Lire (A) 2) Ecrire ("B = "), Lire (B) 3) $SUP \leftarrow (A + B + \text{abs}(A - B)) \text{ div } 2$ 4) $INF \leftarrow (A + B - \text{abs}(A - B)) \text{ div } 2$ 5) Ecrire ("Valeur sup = ", SUP, " Valeur inf = ", INF) 6) Fin Sup_Inf	

Fenni Salah

LES STRUCTURES CONDITIONNELLES

Exercice 1

```

Program Min2 ;
Uses Wincrt ;
Var a, b, min : Integer ;
Begin
  Writeln ('Saisir deux entiers : ') ;
  Readln (a, b) ;
  IF a<b Then min := a
    Else min := b ;
  Writeln ('La plus petite valeur est : ', min) ;
End.
  
```

Exercice 2

```

Program Max3 ;
Uses Wincrt ;
Var a, b, c, maxi : Integer ;
Begin
  Writeln ('Saisir trois entiers : ') ;
  Readln (a, b, c) ;
  maxi := a ;
  IF b>maxi Then maxi := b ;
  IF c>maxi Then maxi := c ;
  Writeln ('La plus grande valeur est : ', maxi) ;
End.
  
```

Exercice 3

```

Program Racine ;
Uses Wincrt ;
Var x : Real ;
Begin
  Write ('Saisir un réel ') ; Readln (x) ;
  IF x >= 0
  Then Writeln ('Racine carrée ', x, ' = ', sqrt(x))
  Else Writeln ('Donnée incorrecte') ;
End.
  
```

Exercice 4

```

Program Aire_Triangle ;
Uses Wincrt ;
Var a, b, c, Surf, P : Real ;
Begin
  Writeln ('Donner 3 réels positifs :'); Readln (a, b, c);
  IF (a+b=c) Or (a+c=b) Or (b+c=a)
  Then Writeln ('Il ne s'agit pas d'un triangle')
  Else Begin
    P := (a+b+c)/2;
    Surf := sqrt (P*(P-a)*(P-b)*(P-c));
    End;
  Writeln ('Aire de triangle = ', Surf:4:2);
End.
  
```

Exercice 6

```

Program Parite ;
Uses Wincrt ;
Var N : Integer ;
Begin
  Writeln ('Donner un entier ') ; Readln (n) ;
  IF n mod 2 = 0
  Then Writeln (n, ' est pair')
  Else Writeln (n, ' est impair') ;
End.
  
```

Exercice 5

```

Program Abs_diff ;
Uses Wincrt ;
Var a, b, z : Integer ;
Begin
  Write ('Donner deux entiers : '); Readln (a, b);
  IF (a-b) < 0 Then z:=b-a
    Else z:=a-b;
  Writeln ('valeur absolue de a-b = ', z);
End.
  
```

Exercice 7

```

Program Chez_la_fourmi ;
Uses Wincrt ;
label 1,2 ;
Var na, nb, err : Integer ;
  a, b, z : Char ;
Begin
  1: Writeln ('Nombre de doigts montrés par le joueur A');
  a:=readkey;
  Writeln ('Nombre de doigts montrés par le joueur B');
  
```

```

  b:=readkey;
  VAL (a, na, err);
  VAL (b, nb, err);
  IF (na+nb) mod 2 =0
  Then Writeln ('Le joueur A gagne.')
  Else Writeln ('Le joueur B gagne. ');
  Writeln ('Voulez vous jouer encore ? (o/n)'); Readln (z);
  IF z='n' Then goto 2 Else goto 1 ;
  2:End.
  
```

Exercice 9

```

Program Sup_Inf ;
Uses Wincrt ;
Var a, b : Integer ;
  sie : String ;
Begin
  Writeln ('Saisir deux entiers : ') ;
  Readln (a, b) ;
  IF a>b
  Then sie := ' est supérieur à '
  Else IF a<b
  Then sie := ' est inférieur à '
  Else sie := ' est égal à ' ;
  Writeln (a, sie, b) ;
End.
  
```

Exercice 8

```

Program Invite ;
Uses Wincrt ;
Var titre, foulon, term1, term2 : String ;
Begin
  Write ('Titre = ') ; Readln (titre) ;
  Write ('Votre nom = ') ; Readln (foulon) ;
  IF titre = 'Mr'
  Then Begin
    term1 := 'e' ;
    term2 := " " ;
  End
  Else IF (titre = 'Mlle') Or (titre='Mme')
  Then Begin
    term1 := 'a' ;
    term2 := 'e' ;
  End ;
  Writeln (titre, ' ', foulon, ' ', soyez 'l', term1,
    ' bienvenu', term2) ;
End.
  
```


Exercice 10

```

Program Ordre ;
Uses Wincrt ;
Var e1, e2, petit, grand : Integer ;
Begin
  Writeln ('Saisir deux entiers : '); Readln (e1, e2) ;
  petit := e1 ;
  grand := e2 ;
  IF e1>e2 Then Begin
    petit := e2 ;
    grand := e1 ;
  End;
  Writeln (petit, grand:3) ;
End.

```

Exercice 11

```

Program Tri;
Uses Wincrt;
Var a, b, c : Integer;
(*****)
Procedure permut (Var x, y : Integer);
Var aux : Integer;
Begin
  aux:=x;
  x:=y;
  y:=aux;
End;
(*****)

```

Exercice 13

```

Program Equa_2d ;
Uses Wincrt ;
Var a, b, c, delta : Real ;
Begin
  Write ('a = '); Readln (a) ;
  Write ('b = '); Readln (b) ;
  Write ('c = '); Readln (c) ;
  IF a = 0 {équation 1er degré}
  Then IF b = 0
    Then IF c = 0
      Then Writeln ('IR')
      Else Writeln ('{ }')
    Else Writeln ('x = ', -c/b)
  Else delta := sqr (b) - 4*a*c ;
  IF delta = 0 {solution réelle double}
  Then Writeln ('x1=x2= ', -b/ (2*a))
  Else IF delta > 0 {deux solutions réelles}
  Then Begin
    Writeln ('x1= ', (-b-sqrt (delta))/ (2*a)) ;
    Writeln ('x2= ', (-b+sqrt (delta))/ (2*a)) ;
  End
  Else Writeln ('Deux solutions complexes') ;
End.

```

Exercice 15

```

Program Touche ;
Uses Wincrt;
Var c : Char ;
nature : String;
Begin
  Writeln ('Taper sur une touche'); Readln (c);
  Case c of
    'a'..'z','A'..'Z' : IF UPCASE(c) in ['A','E','I','U','O','Y']
      Then nature := 'Voyelle'
      Else nature := 'Consonne';
    '0'..'9' : nature := 'Chiffre';
    Else nature := 'Symbole';
  End;
  Writeln (nature);
End.

```

Exercice 12

```

Program Equa_1d ;
Uses Wincrt ;
Var a, b : Real ;
Begin
  Write ('a = '); Readln (a) ;
  Write ('b = '); Readln (b) ;
  IF a <> 0
  Then Writeln ('x = ', -b/a)
  Else IF b = 0
    Then Writeln ('IR')
    Else Writeln ('{ }') ;
End.

```

Begin

```

  Write ('a = '); Readln (a) ;
  Write ('b = '); Readln (b) ;
  Write ('c = '); Readln (c) ;
  IF a>b Then permut (a, b);
  IF b>c Then permut (b, c);
  IF a>b Then permut (a, b);
  Writeln (a, b:4, c:4);
End.

```

Exercice 14

```

Program Inequation ;
Uses Wincrt ;
Var a, b : Real ;
Begin
  Write ('a = '); Readln (a) ;
  Write ('b = '); Readln (b) ;
  IF a>0
  Then Writeln ('x < ', -b/a)
  Else IF a<0
    Then Writeln ('x > ', -b/a)
    Else IF b<0
      Then Writeln ('IR')
      Else Writeln ('Impossible') ;
End.

```

Exercice 16

```

Program Calculette ;
Uses Wincrt ;
Var a, b : Real ;
op : Char ;
Begin
  Readln (a) ; Readln (op) ; Readln (b) ;
  Case op of
    '+' : Writeln (a:3:2, ' ', op, ' ', b:3:2, ' = ', a+b:3:2) ;
    '-' : Writeln (a:3:2, ' ', op, ' ', b:3:2, ' = ', a-b:3:2) ;
    '*' : Writeln (a:3:2, ' ', op, ' ', b:3:2, ' = ', a*b:3:2) ;
    '/' : IF b = 0
      Then Writeln ('impossible')
      Else Writeln (a:3:2, ' ', op, ' ', b:3:2, ' = ', a/b:3:2);
    Else Writeln ('Opérateur incorrect');
  End ;
End.

```

Exercice 17

```

Program Bulletin ;
Uses WinCRT ;
Var moy : Real ;
    me, dec : String ;
Begin
  Write ('Donner la moyenne annuelle : ') ; Readln (moy) ;
  IF moy < 10
  Then Begin
    dec := 'Redouble' ;
    me := '' ;
  End
Else Begin
  dec := 'Admis' ;
  IF moy < 12
  Then me := 'Passable'
  Else IF moy < 14
  Then me := 'Assez bien'
  Else IF moy < 16
  Then me := 'Bien'
  Else IF moy < 18
  Then me := 'Très bien'
  Else me := 'Excellent' ;
  End ;
  Writeln ('Moyenne = ', moy, ' Décision = ', dec,
    ' Mention = ', me) ;
End.

```

Exercice 20

```

Program Anciennete ;
Uses WinCRT ;
Var ji, mi, ai, jf, mf, af, jj, mm, aa : Integer ;
Begin
  Write ('Donner la date initiale : ') ;
  Readln (ji, mi, ai) ;
  Write ('Donner la date finale : ') ;
  Readln (jf, mf, af) ;
  IF ji > jf
  Then Begin
    jf := jf + 30 ;
    mf := mf - 1 ;
  End ;
  IF mi > mf
  Then Begin
    mf := mf + 12 ;
    af := af - 1 ;
  End ;
  jj := jf - ji ;
  mm := mf - mi ;
  aa := af - ai ;
  Writeln (aa, ' années ', mm, ' mois ', jj, ' jours') ;
End.

```

Exercice 21

```

Program Lendemain;
Uses WinCRT;
Var j, m, a : Integer;
Begin
  Writeln ('Donner une date sous forme jj mm aa');
  Readln (j, m, a);
  Case m Of
    1, 3, 5, 7, 8, 10, 12 : IF j<31
      Then j:=j+1
      Else IF m = 12
        Then Begin
          j:=1;
          m:=1;
          a:=a+1;
        End

```

Exercice 18

```

Program nbr_jours_mois;
Uses winCRT;
Var nbj, mm, an : integer;
Begin
  Write ('N° du mois : ') ; readln (mm) ;

  nbj :=31;

  if mm in [4,6,9,11]
  then nbj :=30
  else begin
    write ('Année : '); readln (an);
    nbj := 28;
    if (an mod 4=0)and((an mod 100<>0)or(an mod 400=0))
    then nbj := 29;
  end;

  writeln ('Le nombre de jours du mois saisi est : ', nbj);
End.

```

Exercice 19

```

Program Date ;
Uses WinCRT ;
Var mm, err : Integer ;
    date, jj, aa, mois : String [10] ;
Begin
  Writeln ('Saisir une date sous la forme jj/mm/aaaa') ;
  Readln (date) ;
  jj := COPY (date, 1, 2) ;
  aa := COPY (date, 7, 4) ;
  VAL (COPY (date, 4, 2), mm, err) ;
  Case mm of
    1 : mois := 'Janvier' ;
    2 : mois := 'Février' ;
    3 : mois := 'Mars' ;
    4 : mois := 'Avril' ;
    5 : mois := 'Mai' ;
    6 : mois := 'Juin' ;
    7 : mois := 'Juillet' ;
    8 : mois := 'Août' ;
    9 : mois := 'Septembre' ;
    10 : mois := 'Octobre' ;
    11 : mois := 'Novembre' ;
    12 : mois := 'Décembre' ;
  End ;
  Writeln (jj, mois:2, aa:2) ;
End.

```

```

Else Begin
  j:=1;
  m:=m+1;
End;
2 : IF a mod 4 =0
  Then IF j<29
    Then j:=j+1
    Else Begin
      j:=1;
      m:=3;
    End
  Else IF j<28
    Then j:=j+1
    Else Begin
      j:=1;
      m:=3;

```

```

Else Begin
    j:=1;
    m:=m+1;
    End;
    Writeln ('La date du lendemain est : ',j,'/',m,'/',a);
    End.
4, 6, 9,11: IF j<30
    Then j:=j+1

```

Exercice 22

```

Program Toute_Lettre;
Uses Wincrt;
Const
    Chiffres : Array[0..19] of String = ('','un','deux','trois','quatre','cinq','six','sept','huit','neuf','dix',
    'onze','douze','treize','quatorze','quinze','seize','dix-sept','dix-huit','dix-neuf');
    Dizaines : Array[2..9] of String = ('vingt','trente','quarante','cinquante','soixante','','quatre-vingt','');
Var
    n : Integer;
    result : String;
Begin
    Writeln ('Donner un entier entre 0 et 99'); Readln (n);
    Case n of
        0..19 : result := Chiffres [n];
        20..69,80..89: IF ((n mod 10 = 1) and (n<>81))
            Then result := Dizaines [n div 10] + ' et ' + Chiffres [n mod 10]
            Else result := Dizaines [n div 10] + '' + Chiffres [n mod 10];
        70..79,90..99: IF (n = 71)
            Then result := Dizaines [n div 10 -1] + ' et ' + Chiffres [n mod 10 + 10]
            Else result := Dizaines [n div 10 -1] + '' + Chiffres [n mod 10 + 10];
    End;
    IF n=0 Then Writeln ('zéro')
    Else Writeln (result);
End.

```

Exercice 23

```

Program jour_semaine;
Uses wincrt;
Var
    day, month, year, dayyear, daymonth, weekday, cm:integer;
    jj:string;
Begin
    writeln ('Donner le jour'); readln (day);
    writeln ('Donner le mois'); readln (month);
    writeln ('Donner l''année'); readln (year);
    dayyear:=(year-1)*365 + ((year-1) div 4);
    daymonth:=0;
    for cm:=1 to (month-1) do
        case cm of
            1, 3, 5, 7, 8, 10, 12 : daymonth:=daymonth+31;
            4, 6, 9, 11 : daymonth:=daymonth+30;
            2 : if (year mod 4=0) and ((year mod 100<>0) or (year mod 400 =0))
                then daymonth:=daymonth+29
                else daymonth:=daymonth+28;
        end;
    weekday:=(dayyear+daymonth+day) mod 7;
    case weekday of
        0:jj:='Dimanche';
        1:jj:='Lundi';
        2:jj:='Mardi';
        3:jj:='Mercredi';
        4:jj:='Jeudi';
        5:jj:='Vendredi';
        6:jj:='Samedi';
    end;
    writeln ('Le jour correspondant est ', jj);
End.

```

LES STRUCTURES ITERATIVES

Exercice 1

```

Program Alphabet;
Uses WinCRT;
Var c : Char;
Begin
  FOR c:= 'A' To 'Z' Do Write (c:2);
  Writeln;
  FOR c:= 'Z' Downto 'A' Do Write (c:2);
End.
  
```

Exercice 2

```

Program Table3;
Uses WinCRT;
Const n = 10;
Var i : Integer;
Begin
  FOR i:= 1 To n Do
    Writeln ('3*',i,' = ',3*i);
End.
  
```

Exercice 3

```

Program Somme_Produit;
Uses WinCRT;
Var s, i : Integer; p : Real;
Begin
  S := 0; P := 1;
  FOR i:= 1 To 20 Do
    Begin
      s := s + i;
      p := p * i;
    End;
  Writeln ('Somme = ', s);
  Writeln ('Produit = ', p:2:2);
End.
  
```

Exercice 4

```

Program Suite;
Uses WinCRT;
Var som, i, u : Integer;
Begin
  som := 0;
  u := 2;
  FOR i := 1 To 100 Do
    Begin
      som := som + u;
      u := u + 3;
    End;
  Writeln (som);
End.
  
```

Exercice 4

```

Program Suite;
Uses WinCRT;
Var som, i, u : Integer;
Begin
  som := 0;
  u := 2;
  i := 1;
  Repeat
    som := som + u;
    u := u + 3;
    i := i+1;
  Until (i>100);
  Writeln (som);
End.
  
```

Exercice 4

```

Program Suite;
Uses WinCRT;
Var som, i, u : Integer;
Begin
  som := 0;
  u := 2;
  i := 1;
  While (i<=100) Do
    Begin
      som := som + u;
      u := u+3;
      i := i+1;
    End;
  Writeln (som);
End.
  
```

Exercice 5

```

Program Pythagore;
Uses WinCRT;
Const n = 9;
Var i, j : Byte;
Begin
  FOR i:=1 To n Do
    Begin
      FOR j:=1 To n Do Write (i * j : 4);
      Writeln;
    End;
End.
  
```

Exercice 6

```

Program Pyramide;
uses winCRT;
const N=4;
var i, j : integer;
begin
  for i:=0 to n do
    begin
      for j:=i+1 to n do write (' ');
      for j:=-i to i do write ('*');
      writeln;
    end;
end;
end.
  
```

Exercice 7

```

Program Moy_Notes;
Uses WinCRT;
Var i, n : Integer;
    note, s : Real;
Begin
  Write ('Combien de notes : '); Readln (n);
  s:=0;
  FOR i:=1 To n Do
    Begin
      Write ('Note ', i, ' : ');
      Readln (note);
      s := s+note;
    End;
  Writeln ('Moyenne de ces ', n, ' notes : ', s/n:2:2);
End.
  
```

Exercice 8

```

Program Factoriel;
Uses WinCRT;
Var i, n : Byte;
    fact : Real;
Begin
  Repeat
    Writeln ('Saisir un entier');
    Readln (n);
  Until n IN [0..255];

  fact := 1;
  FOR i := 2 To n Do fact := fact * i;

  Writeln (n, ' ! = ', fact);
End.
  
```

<p>End.</p> <p>Exercice 10 Program Diviseurs; Uses Wincrt; Var n, m, r : Integer; Begin Writeln ('Donner un entier'); Readln (n); m:=n; Repeat r:= m mod 10; m:= m div 10; IF (n mod r = 0) Then Write (r, ' '); Until m=0; End.</p>	<p>End.</p> <p>Exercice 11 Program Som_15; Uses Wincrt; Var i, j, k : Integer; Begin FOR i:=1 TO 9 DO FOR j:=1 TO 9 DO FOR k:=1 TO 9 DO IF (i+j+k=15) Then Begin Writeln (i, ' ', j, ' ', k); Readln ; End; End.</p>
<p>Exercice 9 Program Jeu ; Uses Wincrt ; Label 1, 2 ; Var np, nc, essai : Integer ; z : Char ; Begin 1: Clrscr ; Randomize ; nc := Random (100) +1 ; essai := 0 ; Repeat essai := essai+1 ; Write ('Essai numéro ', essai, 'Votre nombre : ':20); Readln (np) ;</p>	<p>IF np > nc Then Writeln ('C'est grand') Else IF np < nc Then Writeln ('C'est petit') Else Writeln ('Bravo vous avez gagné !!') ; Until (np = nc) Or (essai = 7) ; IF np<>nc Then Writeln ('Perdu, le nombre cherché est : ', nc); Writeln ('Voulez vous jouer encore ? (o/n)'); Readln (z); IF z='n' Then goto 2 Else goto 1 ; 2:End.</p>
<p>Exercice 12 Program Histogramme; Uses Wincrt; Var a, b, c, max, i : Integer; Begin Writeln ('Entrer trois entiers compris entre 0 et 15'); Readln (a, b, c); max:=a; IF b>max Then max:=b; IF c>max Then max:=c; FOR i:= max Downto 1 Do Begin IF i>a Then Write ('') Else Write ('A'); IF i>b Then Write ('':4) Else Write ('B':4); IF i>c Then Writeln ('':4) Else Writeln ('C':4); End; End.</p>	<p>Program histogramme; Uses wincrt; Var n1, n2, n3:integer; Procedure lecture (var n:integer); Begin writeln ('Entrer trois entiers compris entre 0 et 15'); repeat readln(n) until n in [0..15] ; End; Procedure affiche (n,c:integer; ca:char); Var l,i:integer ; Begin l:=21; {numéro de ligne} for i:=1 to n do begin gotoxy(c,l); writeln(ca); l:=l-1; end; End; Begin lecture(n1); lecture(n2); lecture(n3); affiche(n1,10,'A'); affiche(n2,14,'B'); affiche(n3,18,'C'); End.</p>
<p>Exercice 13 Program Som_Chiffres; Uses Wincrt; Var n, som, r : Integer; Begin Writeln ('Donner un entier'); Readln (n); som:=0; Repeat r:= n mod 10; som:=som+r; n:= n div 10; Until n=0; Writeln ('La somme de chiffres est : ', som); End.</p>	<p>Exercice 14 Program Nbr_Cube; Uses Wincrt; Var k, c, d, u : Integer; Begin FOR k:=100 To 999 Do Begin c:=k div 100 ; d:=(k div 10) mod 10; u:=k mod 10 ; IF (u*u*u+d*d*d+c*c*c) = K Then Writeln (k, ' est un nombre cubique'); End; End.</p>

Exercice 15

```

Program Somme ;
Uses WinCRT;
Var n, i : Integer ;
    s1, s2, s3 : Real ;
Begin
  Repeat
    Write ('Saisir un entier impair : '); Readln (n);
  Until odd (n);
  s1:=0; s2:=0; s3:=0;
  FOR i:=1 To n Do
    IF odd (i) Then s2 := s2 + 1/i
                Else s3 := s3 - 1/i ;
    s1:= s2 - s3 ;
    Writeln (s1:8:2, s2:8:2, s3:8:2);
  End.

```

Exercice 16

```

Program Syracuse ;
Uses WinCRT;
Var n, i, s : Integer ;
Begin
  Writeln ('Saisir un entier > 0 '); Readln (n);
  S:=n;
  FOR i:=1 To 50 Do
    Begin
      Write (S, ' ');
      IF S mod 2 = 0
        Then S:= S div 2
        Else S:= 3*S+1;
    End;
  End.

```

Exercice 17

```

Program Probabilite;
Uses WinCRT ;
Const n = 12; essai = 100;
Var d1, d2, d3, cumul, i : Byte;
Begin
  Randomize;
  cumul :=0;
  FOR i:=1 To essai Do
    Begin
      d1 := 1 + Random (6);
      d2 := 1 + Random (6);
      d3 := 1 + Random (6);
      IF (d1+d2+d3 = n) Then cumul := cumul +1;
    End;
  Writeln ('Probabilité est : ', cumul/essai :5:2);
End.

```

Exercice 18

```

Program Puissance_n ;
Uses WinCRT ;
Var n, k : Integer ;
    y, x : Real ;
Begin
  Write ('Saisir un nombre réel : '); Readln (x) ;
  Write ('Saisir la puissance n : '); Readln (n) ;
  y := 1 ;
  FOR k := 1 To abs (n) Do y := y * x ;
  IF n<0 Then y := 1/y ;
  Writeln (x:5:2, ' puissance ', n, ' = ', y:5:2) ;
End.

```

Exercice 19

```

Program Multiplication_Addition ;
Uses WinCRT ;
Var x, y, s, aux, i : Integer ;
Begin
  Writeln ('Donner deux entiers'); Readln (x, y) ;
  Write (x, ' * ', y, ' = ');
  IF abs(y)>abs(x) Then Begin
    aux := x ;
    x := y ;
    y := aux ;
  End ;
  IF y<0 Then Begin
    y:= -y ;
    x:= -x ;
  End ;
  s := 0 ;
  FOR i:=1 To y Do s := s+x ;
  Writeln (s) ;
End.

```

Exercice 20

```

Program Suite ;
Uses WinCRT ;
Var i, n : Integer ; s, invfact : Real ;
Begin
  Writeln ('Donner un entier'); Readln (n) ;
  s := 1 ;
  invfact := 1 ;
  FOR i := 1 To n Do
    Begin
      invfact := invfact/i ;
      s := s + invfact ;
    End ;
  Writeln (s:5:2) ;
End.

```

Exercice 21

```

program produits;
uses winCRT;
var a, b, c, d:integer;
begin
  for a:=1 to 9 do
    for c:=a to 9 do
      for b:=c downto a do
        for d:=c downto a do
          if ((10*a+b)*(10*c+d) = (10*b+a)*(10*d+c))
            and (a<>b) and (b<>c)
          then Writeln (a, b, ' * ', c, d, ' = ', b, a, ' * ', d, c) ;
        end.

```

Exercice 22

```

PROGRAM PI_WALLIS;
USES WINCRT;
VAR P, r, diff : Real;
    i : LONGINT;
BEGIN
  P:=1; i:=0;
  REPEAT
    i:=i+2;
    r:=i/(i-1)*i/(i+1);

```

Exercice 23

```

Program PGCD_Euclide;
Uses WinCRT;
Var a, b, r : Integer;
Begin
  Repeat
    Writeln ('Saisir deux entiers > 0'); Readln (a, b);
  Until (a>0) and (b>0) ;
  While b<>0 Do
    Begin

```



```

diff:= (P*r) - P;
P:=P*r;
UNTIL abs (diff) < 1e-8;
WRITELN ('Par la formule de Wallis Pi = ', 2*P:2:7);
END.

```

Exercise 25

```

Program PPCM;
Uses Wincrt;
Var pcm, m, n, aux : Integer;
Begin
  Repeat
    Writeln ('Saisir deux entiers > 0'); Readln (m, n);
  Until (m>0) and (n>0);
  IF m < n Then Begin
    aux:= m;
    m := n;
    n := aux;
  End;
  pcm := m;
  While (pcm mod n <> 0) Do pcm := pcm + m;
  Writeln ('PPCM = ', pcm);
End.

```

Exercise 26

```

Program Fibonacci;
uses wincrt;
var k, f0, f1, f2 : integer;
begin
  f0 := 1; f1 := 1;
  write (f0, ' ', f1, ' ');

  for k := 2 to 19 do
  begin
    f2 := f1+f0;
    f0 := f1;
    f1 := f2;
    write (f2, ' ');
  end;
end.

```

Exercise 28

```

Program Parfait;
uses wincrt;
var nb, d, som, a, b : integer;
begin
  repeat
    Readln(a,b);
  until (1<a) and (a<b);
  for nb:=a to b do
  begin
    som:=0;
    for d:=1 to (nb div 2) do
      if (nb mod d = 0) then som:=som+d;
    if nb=som then writeln (nb, ' est parfait');
  end;
end.

```

Exercise 29

```

Program Amis;
uses wincrt;
var m, n, sdn, sdm : integer;
(*****)
function diviseurs (x : integer) : integer;
var sdx, i : integer;
begin
  sdx:=1;
  for i:=2 to (x div 2) do
    if (x mod i) = 0 then sdx := sdx+i;
  diviseurs:=sdx;
end;
(*****)
begin
  for m:=1 to 1000 do
    for n:=1 to 1000 do

```

```

r := a mod b;
a := b;
b := r;
End;
Writeln ('PGCD = ', a);
End.

```

Exercise 24

```

Program PGCD_Diff;
Uses Wincrt;
Var a, b : Integer;
Begin
  Repeat
    Writeln ('Saisir deux entiers >0 '); Readln (a, b);
  Until (a>0) and (b>0);
  While a<>b Do
    IF a>b Then a:= a-b
    Else b:= b-a;

  Writeln ('PGCD = ', a);
End.

```

Exercise 27

```

Program Nbre_Premiers;
uses wincrt;
var nb, i : integer;
begin
  for nb := 2 to 400 do
  begin
    i := 2;
    while (nb mod i <> 0) and (i <= nb div 2) do i:= i+1;
    if (i > nb div 2) then write (nb:4);
  end;
end.

```

Exercise 32

```

Program Exponentiel;
uses wincrt;
var x, s, epsilon, p, f:real;
i:integer;
begin
  write ('epsilon = '); readln (epsilon);
  write ('x = '); readln (x);
  s:=1; i:=1; p:=1; f:=1;
  repeat
    p:=p*x;
    f:=f*i;
    s:=s+p/f;
    i:=i+1;
  until abs(p/f) <= epsilon;
  writeln ('expn = ', s:2:10);
end.

```

Exercise 30

```

Program Facteur_Premier;
uses wincrt;
type tab = array [1..100] of integer;
var fp : tab;
n, i, f : integer;
begin
  repeat
    writeln ('donner un entier entre 2 et 1000'); readln (n);
  until (n>=2) and (n<=1000);
  write (n, ' ');
  i := 2; f:=0;
  repeat
    if (n mod i = 0)
    then begin
      n:= n div i;
      f:=f+1;

```

```

begin
  sdn := diviseurs (n);
  sdm := diviseurs (m);
  if (sdm=n) and (sdn=m)
    then writeln (n, ' et ', m, ' sont amis' );
end;
end.

```

```

fp[f] := i;
end
else i:=i+1;
until (n=1);
write (fp[1]);
for i:=2 to f do write ('*', fp[i]);
end.

```

Exercise 31

```

Program calcul_sinus;
uses wincrt;
var x:real;
(*****)
function sinus (x:real) : real;
var som, term1, term2, i:real;
begin
  som:=x;
  term2:=x;
  i:=1;
  repeat
    i:=i+2;
    term1:=term2;
    term2:=term2 * -sqr(x) / (i*(i-1));
    som:=som+term2;
  until abs (term2-term1) <= 0.0001;
  sinus:=som;
end;
(*****p.p*****)
begin
  repeat
    write ('donner un réel x ');
    readln(x);
  until (- Pi <= x) and (x <= Pi);
  write ('sin(', x:1:2,') = ', sinus(x):10:10);
end.

```

Exercise 33

```

Program Somme;
Uses Wincrt;
Var i, n : Integer; s : Real;
(*****)
Procedure saisie (Var m : Integer);
Begin
  Repeat
    Writeln ('Donner un entier positif');
    Readln (m);
  Until m>0;
End;
(*****)
Function puissance (x : Integer):Longint;
Var j : Integer ; p : Longint;
Begin
  p:=1;
  FOR j:=1 To x Do p:=p*x;
  puissance:=p;
End;
(*****)
Begin
  saisie (n);
  s:=0;
  FOR i:=1 To n Do s:= s + (2*i-1) / puissance (i);
  Writeln ('la somme = ', s:2:3);
End.

```

Exercise 34

```

Program Combinaison ;
Uses Wincrt ;
Var cnp : Real ;
n, p : Integer ;
(*****)
Function Fact (x : Integer) : LongInt ;
Var f : LongInt ; i : Integer ;
Begin
  f := 1 ;
  FOR i := 2 To x Do f := f * i ;
  fact := f ;
End ;
(*****)
Begin
  Repeat
    Writeln ('Donner deux entiers : ');
    Readln (p, n);
  Until (0<p) and (p<n) ;
  cnp := fact (n) / (fact (p) * fact (n-p)) ;
  Writeln ('Combinaison = ', cnp :4:2) ;
End.

```

Exercise 34

```

Program Combinaison ;
Uses Wincrt ;
Var cnp : Real ; n, p, i : Integer ;
Nf, pf, npf : Longint ;
Begin
  Repeat
    Write ('p = '); Readln (p);
    Write ('n = '); Readln (n);
  Until (0<p) and (p<n) ;
  Nf :=1;
  Pf :=1;
  Npf :=1;
  FOR i:=2 To n Do
    Begin
      Nf := nf*i ;
      IF i<=p Then pf := pf*i;
      IF i<=n-p Then npf := npf*i;
    End;
  Cnp := nf / (pf*npf) ;
  Writeln ('Combinaison = ', cnp :4:2) ;
End.

```

Exercise 35

```

Program Ordered ;
Uses Wincrt;
Var n : Integer;
i, mn, mx, p : Longint;
(*****)
Procedure min_max (m : Integer; Var min, max:Longint);
Var i : Integer;
Begin
  min:=0; max:=0;
  FOR i:=1 To m Do
    Begin
      min:=10*min + i;
      max:=10*max + (9-m+i);
    End;
End;
(*****)

```

```

ordre:=valide;
End;
(*****)
Procedure saisie (Var m : Integer);
Begin
  Repeat
    Write ('Donner un entier n compris entre 2 et 8 : ');
    Readln (m);
  Until m in [2..8];
End;
(*****)
Begin
  saisie (n);
  min_max (n, mn, mx);
  p:=0;
  FOR i:=mn To mx DO
    IF ordre (i) Then

```

```

Function ordre (m:Longint) : Boolean;
Var c1, c2 : Integer; valide : Boolean;
Begin
  Repeat
    c1:= m MOD 10;
    m:= m DIV 10;
    c2:= m MOD 10;
    valide:=(c1>c2);
  Until Not (valide) Or (m<10) ;

```

```

Begin
  p:=p+1;
  Writeln (p:10, ' - ', i); Readln ;
End;
End.

```

Exercice 36

```

Program Nbre_Impairs;
Uses Wincrt;
Var i, n : Integer;
Begin
  n:=0;
  FOR i:=1 To 99 Do
    IF ODD (i) and (i mod 7 <>0)
      Then Begin
        n:=n+1;
        IF n mod 5 <>0
          Then Write (i:4)
          Else Writeln (i:4);
      End;
  End;
End.

```

Exercice 37

```

program sommes_entiers;
uses wincrt;
var n, i, j, k, s:integer;
begin
  writeln ('entrer la valeur de N :'); readln (n);
  for i:=1 to n div 2 do
    begin
      S := i; j:= i;
      Repeat
        J := j +1;
        S := S + j;
      until s >= n;
      if S=N then
        begin
          write (n, ' = ',i);
          for k:=i+1 to j do write(' + ',k);
          writeln;
        end;
      end;
    end;
end.

```

Exercice 38

```

Program Reine ;
Uses Wincrt;
Var x, y, i, j : Byte ;
Begin
  Write ('Les coordonnées de la dame: X = '); Readln (X);
  Write (' Y = '); Readln (Y);
  FOR i:=1 To 8 Do
    Begin
      FOR j:=1 To 8 Do
        IF (i=x) and (j=y)
          Then Write (' R ')
          Else IF (i=x) Or (j=y) Or (abs(x-i)=abs(y-j))
            Then Write (' * ')
            Else Write (' ');
      Writeln ;
    End;
  End.

```

Exercice 39

```

Program ppcm_pgcd;
uses wincrt;
var a, b : integer;
(*****)
Procedure saisie (var a,b:integer);
begin
  repeat
    writeln('Donner deux entiers >0');
    readln (a, b);
  until (a>0) and (b>0);
end;
(*****)
Procedure affiche(a,b:integer);
var k:integer;
begin
  k := 0;
  repeat
    k := k+1
  until (a*k) mod b = 0;
  writeln ('ppcm de ', a, ' et ', b, ' = ', a*k);
  writeln ('pgcd de ', a, ' et ', b, ' = ', b div k);
end;
(***** p,p *****)

```

Exercice 40

```

Program produit_ab;
uses wincrt;
var a, b,ppcm,pgcd : integer;
(*****)
Procedure pgcd_ppcm(a,b:integer;var pgcd,ppcm:integer);
var k:integer;
begin
  k := 0;
  repeat
    k := k+1
  until (a*k) mod b = 0;
  pgcd:=a*k;
  ppcm:=b div k;
end;
(***** p,p *****)
begin
  writeln('Donner a et b : ');
  readln (a, b);
  pgcd_ppcm(a,b,pgcd,ppcm);
  writeln(a, ' * ',b, ' = ',pgcd*ppcm);
end.

```

```
BEGIN
  saisie(a,b);
  affiche(a,b);
END.
```

Exercise 41

```
Program Exercise_41;
uses wincrt;
var a,b,s:real;
    n,i,signe:integer;
(*****)
Procedure saisie(var a,b:real;var n:integer);
begin
  writeln('donner a et b'); readln(a,b);
  repeat
    writeln('donner n ');
    readln(n);
  until (n>=3) and odd(n);
end;
(*****)
Function power ( x:real;n:integer):real;
var k:integer;
    p:real;
begin
  p:=1;
  for k:= 1 to n do p:=p*x;
  power:=p;
end;
(***** p,p *****)
begin
  saisie(a,b,n);
  s:=0;
  signe:=1;
  for i:=0 to n-1 do
    begin
      s:=s+signe*power(b,i)*power(a,n-i-1);
      signe:=-signe;
    end;
  writeln((a+b)*s:2:2);
end.
```

Exercise 43

```
Program somme ;
Uses WinCrt ;
Var n,p,signe:Integer;
    som:real;
(*****)
Function Comb(p,n:Integer):real;
(*****)
Function Fact (x:Integer):LongInt;
var f:LongInt; i:integer;
begin
  f:=1;
  for i:=2 to x do f:=f*i;
  fact:=f;
end;
(*****)
begin
  comb:=fact(n)/(fact(p)*fact(n-p));
end;
(***** p,p *****)
Begin
  Writeln ('Donner n : ');
  ReadLn (n);
  som:=1;
  signe=-1;
  for p:=1 to 2*n do
    begin
```

Exercise 42

```
Program divisible_11;
uses wincrt;
var x:integer;
(*****)
Procedure saisie (var x:integer);
begin
  repeat
    write('Donner un entier ');
    readln(x);
  until x>0;
end;
(*****)
Function divs_11(x:integer):boolean;
var signe,som:integer;
begin
  signe:=1;
  som:=0;
  repeat
    som:=som+signe*(x mod 10);
    x:=x div 10;
    signe:=-signe;
  until x=0;
  divs_11:= som mod 11 = 0;
end;
(***** p,p *****)
begin
  saisie(x);
  if divs_11(x)
  then writeln('divisible par 11')
  else writeln('non divisible par 11');
end.
```

Exercise 44

```
Program divis_7_13;
uses wincrt;
var n,nb:integer;
(*****)
Function div_7 ( n : integer ) : boolean ;
begin
  while (n>99) do n := (n div 10) - 2 * (n mod 10) ;
  div_7:=(n mod 7 = 0);
end ;
(*****)
Function div_13 ( n : integer ) : boolean ;
begin
  while (n>99) do n := (n div 10) + 4 * (n mod 10) ;
  div_13:=(n mod 13 =0);
end ;
(***** p,p *****)
BEGIN
  writeln('les nombres divisibles par 7 :');
  nb:=0;n:=0;
  repeat
    if div_7(n) then begin
      write(n:5);
      nb:=nb+1;
    end;
    n:=n+1;
  until nb=100;
```

```

    som:=som+signe*sqr(comb(p,2*n));
    signe:=-signe;
end;

Writeln ('somme = ',som:2:2);
End.

```

```

writeln;
writeln('les nombres divisibles par 13 :');
nb:=0; n:=0;
repeat
    if div_13(n) then begin
        write(n:5);
        nb:=nb+1;
        end;

        n:=n+1;
    until nb=100;
END.

```

Exercice 45

```

Program Exercice45;
uses wincrt;
var k: integer;
(*****)
Function kaprekar(m : longint): boolean;
var l,n1,n2,err : integer;
    ch,ch1,ch2 : string;
begin
    str(sqr(m),ch);
    l := length(ch);
    ch1 := copy(ch, 1, l div 2);
    ch2 := copy(ch, (l div 2)+1, l);
    val(ch1,n1,err);
    val(ch2,n2,err);

    kaprekar := (m=n1+n2);
end;
(***** p,p *****)
Begin
    for k:=1 to 1000 do
        if kaprekar(k) then writeln(k);
    End.

```

FENNI SALAH

FENNI

LES TABLEAUX

Exercise 1

```

Program Som_Produit_MoyArith ;
Uses Wincrt ;
Type Tab = Array [1..10] of Integer ;
Var T : Tab ; n, i, st : Integer ; mt, pt : Real ;
Begin
  Repeat
    Write ('N = ');
    Readln (n) ;
  Until (n>5) and (n<=10) ;
  FOR i := 1 To n Do
    Repeat
      Write ('T', i, ' = ');
      Readln (T[i]) ;
    Until (1<= T[i]) and (T[i] <= 20) ;
    st := 0 ;
    pt := 1 ;
  FOR i := 1 To n Do
    Begin
      St := st + T[i] ;
      Pt := pt * T[i] ;
    End ;
  Writeln ('Somme = ', st);
  Writeln ('Produit = ', trunc (pt));
  Writeln ('Moyenne arithmétique = ', st/n:2:2);
End.

```

Exercise 3

```

Program Freq_Lettre ;
Uses Wincrt ;
Const n=35 ;
Var LET : Array [1..n] of Char ;
    FE : Array ['A'..'Z'] of Byte ;
    i : Byte ; j : Char ;
Begin
  Randomize ;
  FOR i := 1 To n Do
    Begin
      LET[i] := CHR (65+ Random (26)) ;
      Write (LET[i] :2) ;
    End;
  FOR j := 'A' To 'Z' Do FE[j] := 0 ;
  FOR i := 1 To n Do
    FE[LET[i]] := FE[LET[i]] + 1 ;

  Writeln ; Writeln ;
  FOR j := 'A' To 'Z' Do Write (j:2);
  Writeln ;
  FOR j := 'A' To 'Z' Do Write (FE[j]:2) ;
End.

```

Exercise 2

```

Program Affich_sans_redondance;
Uses Wincrt;
Var T : Array [1..20] of Char;
    n, i, j : Integer;
Begin
  Repeat
    Writeln ('Donner un entier'); Readln (n);
  Until n in [3..20] ;
  FOR i:=1 To n Do
    Repeat
      Writeln ('Saisir la case d'ordre ', i);
      Readln (T[i]);
      Until upcase (T[i]) in ['A'..'Z'] ;
  FOR i:=1 To 20 Do Write (T[i], ' ');
  Writeln ;
  Writeln ;
  Write (T[1], ' ');
  FOR i:=2 To 20 Do
    Begin
      j:=i;
      While (j>2) and (T[i]<>T[j-1]) Do j:=j-1;
      IF T[i] <>T[j-1] Then Write (T[i], ' ');
    End;
  End.

```

Exercise 4

```

Program Conversion_base10_base2;
Uses Wincrt;
Var rest : Array [1..50] of 0..1;
    n, i, j : Integer;
Begin
  Repeat
    Writeln ('Donner un entier positif'); Readln (n);
  Until (n > 0);
  i:=0;
  Repeat
    i:=i+1;
    rest[i]:=n mod 2;
    n:=n div 2;
  Until n=0;
  FOR j:=i Downto 1 Do Write (rest[j]);
End.

```

===== Solution 2 =====

```

Program Conversion_base10_base2;
Uses Wincrt;
var n:integer;
(*****
procedure saisir(var n:integer);
begin
  Repeat
    Writeln ('Donner un entier positif'); Readln (n);
  Until (n > 0);
end;
(*****
function dec_bin (n:integer):string;
var chb,chr:string;
    r:0..1;
begin
  chb:="";
  Repeat
    r:=n mod 2;
    str(r, chr);
    insert (chr, chb, 1);
    n:=n div 2;
  Until n=0;
  dec_bin:=chb;
end;
(*****pp*****
Begin
  Saisir (n);

```


	<pre>writeln(',',n,')10 = (',dec_bin(n),')2'); End.</pre>
<p>Exercice 5</p> <pre>Program Conversion_b1_b2; Uses WinCRT; Var nb, reste : Array [1..50] of 0..15; b1, b2, n, i, j, err, nb10 : Integer; nch : String; Begin Repeat Write ('Base b1 = '); Readln (b1); Write ('Base b2 = '); Readln (b2); Until (b1 in [2..16]) and (b2 in [2..16]); Writeln ('Donner le nombre à convertir'); Readln (nch); n:=Length (nch); FOR i:=1 To n Do IF ORD (nch[i]) < 65 Then VAL (nch[i], nb[i], err) Else nb[i] :=(ORD (nch[i]) - ORD ('A') + 10) ;</pre>	<pre>{conversion de la base b1 au décimal} nb10 :=0; FOR i:=1 To n-1 Do nb10:=(nb10+nb[i])*b1; nb10:=nb10+nb[n]; {conversion de nb10 du décimal à la base b2} i:=0; Repeat i:=i+1; reste[i]:=nb10 mod b2; nb10:=nb10 div b2; Until nb10=0; {affichage du résultat} FOR j:=i Downto 1 Do IF reste[j] < 10 Then Write (reste[j]) Else Write (CHR (reste[j]-10 + ORD ('A')));</pre> <p>End.</p>
<p>Exercice 6</p> <pre>Program Eclater_tab ; Uses WinCRT ; Type Tab = Array [1..50] of Integer ; Var T, TN, TP : Tab ; n, i, j, k : Integer ; Begin Repeat Writeln ('Saisir un entier') ; Readln (n) ; Until (n>=10) and (n<=50) ; Writeln ('Saisir les ', n, ' éléments de T') ; FOR i:=1 To n Do Readln (T[i]) ;</pre>	<pre>j := 0 ; k := 0 ; FOR i := 1 To n Do IF T[i] < 0 Then Begin j := j+1 ; TN[j] := T[i] ; End Else Begin k := k+1 ; TP[k] := T[i] ; End ; FOR i := 1 To j Do Write (TN[i]:4) ; Writeln ; FOR i := 1 To k Do Write (TP[i]:4) ;</pre> <p>End.</p>
<p>Exercice 7</p> <pre>Program Inverser_tab ; Uses WinCRT ; Type Tab = Array [1..50] of Integer ; Var T : Tab ; i, n, aux : Integer ; Begin Repeat Writeln ('Saisir un entier') ; Readln (n) ; Until n in [10..50] ; Writeln ('Saisir les ', n, ' éléments de T') ; FOR i:= 1 To n Do Readln (T[i]) ;</pre>	<pre>FOR i := 1 To (n div 2) Do Begin aux := T[i] ; T[i] := T[n-i+1] ; T[n-i+1] := aux ; End ; Writeln ; Writeln ; Writeln ('Tableau inversé :') ; FOR i := 1 To n Do Write(T[i]:4) ;</pre> <p>End.</p>
<p>Exercice 8</p> <pre>Program Regrouper_tab ; Uses WinCRT ; Var T : Array [1..50] of Integer ; i, j, k, n, tmp : Integer ; Begin Repeat Write ('N = '); Readln (n) ; Until (n>=10) and (n<=50) ; Randomize ; FOR i:= 1 To n Do Begin T[i] := -20+Random (41) ; Write (T[i]:4) ; End;</pre>	<pre>k:=0 ; FOR i:= 1 To n Do IF (T[i] mod 2) = 0 Then Begin k := k+1 ; tmp := T[i] ; FOR j:=i Downto k+1 Do T[j]:=T[j-1] ; T[k] := tmp ; End ; Writeln ; Writeln ; FOR i:= 1 To n Do Write (T[i]:4) ;</pre> <p>End.</p>
<p>Exercice 9</p> <pre>Program Min_Max_tab ; Uses WinCRT; Var T : Array [1..50] of Integer; i, min, max, n : Integer; Begin Repeat Readln (n) Until (n>=10) and (n<=50); FOR i:=1 TO N DO Readln (T[i]);</pre>	<pre>min:=T[1]; max:=T[1]; FOR i:=2 TO n DO Begin IF T[i]<min Then min:=T[i]; IF T[i]>max Then max:=T[i]; End; Writeln ('Valeur maximale = ', max);</pre>

```

Writeln ('Valeur minimale = ', min);
End.

```

Exercice 11

```

Program Ranger_tab ;
Uses Wincrt ;
Type Tab = Array [1..10] of Integer ;
Var T, R : Tab ;
    i, j : Integer ;
Begin
    Writeln ('Saisir les 10 éléments de T') ;
    FOR i:=1 To 10 Do
        Readln (T[i]) ;
    j:=0 ;
    FOR i:= 1 To 10 Do
        IF T[i] >= 0 Then Begin
            j := j+1 ;
            R[j] := T[i] ;
        End;
    FOR i:= 1 To 10 Do
        IF T[i] < 0 Then Begin
            j := j+1 ;
            R[j] := T[i] ;
        End;
    FOR i:= 1 To 10 Do Write (T[i]:4) ;
    Writeln ;
    FOR i:= 1 To 10 Do Write (R[i]:4) ;
End.

```

Exercice 10

```

Program Symetri_tab ;
Uses Wincrt ;
Type Tab = Array [1..50] of Integer ;
Var T : Tab ;
    i, j, n : Integer ;
Begin
    Repeat
        Writeln ('Saisir un entier') ;
        Readln (n) ;
    Until (n>1) and (n mod 2 =0) ;
    Writeln ('Saisir ', n div 2, ' éléments de T') ;
    FOR i:= 1 To (n div 2) Do
        Begin
            Readln (T[2*i-1]) ;
            T[2*i] := T[2*i-1];
        End;
    FOR i:= 1 To (n div 2)-1 Do
        Begin
            FOR j:=i+1 To n-i Do T[j] := T[j+1] ;
            T[n-i+1] := T[i] ;
        End ;
    Writeln ('Tableau symétrique :') ;
    FOR i:= 1 To n Do Write (T[i]:4) ;
End.

```

Exercice 12

```

Program Recherche_Dichotomique_tab ;
Uses Wincrt;
type Tab = Array [1..50] of Integer;
Var T : Tab;
    N, V : Integer;
(*****)
Procedure Saisie (Var T : Tab ; Var n, v : Integer);
Var i : Integer;
Begin
    Repeat
        Writeln ('Donner un entier'); Readln (n);
    Until n in [10..50];

    Writeln ('Saisir les éléments de T en ordre croissant');
    Readln (T[1]);
    FOR i:=2 To n Do
        Repeat
            Readln (T[i])
        Until T[i] >= T[i-1];
    Writeln ('Donner la valeur à chercher'); Readln (v);
End;

```

(*****)

```

Function Recherche (v, n : Integer ; T : Tab) : Integer;
Var d, g, m, pos : Integer;
Begin
    g:=1 ; d:=n ; pos:=0;
    Repeat
        m:=(g+d) div 2 ;
        IF V=T[m] Then pos:=m
        Else IF V>T[m]
            Then g:=m+1
            Else d:=m-1;
    Until (pos=m) Or (g>d);
    recherche:= pos ;
End;
(*****)
Begin
    Saisie (T, N, V);
    IF Recherche (V, N, T) = 0
        Then Writeln (V, ' ne figure pas dans le tableau')
        Else Writeln (V, ' se trouve à la position ',
            recherche (V, N, T));
End.

```

Exercice 12

```

Program Recherche_sequentielle_tab ;
Uses Wincrt ;
Var T : Array [1..50] of Integer ;
    i, v, n : Integer ;
Begin
    Repeat
        Write ('N = '); Readln (n);
    Until (10<=n) and (n<=50);
    Writeln ('Saisir les ', n, ' éléments de T');
    FOR i:= 1 To n Do Readln (T[i]) ;
    Writeln ('Donner la valeur à chercher') ; Readln (v) ;
    i := 0 ;
    Repeat
        i :=i+1 ;
    Until (v=T[i]) Or (i=n) ;
    IF v=T[i]
        Then Writeln (v, ' se trouve à la position ', i)
        Else Writeln (v, ' ne figure pas dans le tableau');
End.

```

Exercice 13

```

Program Regrouper_Tab ;
Uses Wincrt ;
Var T : Array [1..20] of Integer ;
    i, j, k, n, tmp : Integer ;
Begin
    Repeat
        Write ('N = '); Readln (n) ;
    Until (n>=2) and (n<=20) ;
    Writeln ('Saisir les éléments de T') ;
    FOR i:=1 To n Do Readln (T[i]) ;
    FOR i:=1 To n-1 Do
        FOR j:=i+1 To n Do
            IF (T[j] = T[i]) Then T[j]:=0 ;
    k:=0 ;
    FOR i:=1 To n Do
        IF T[i]<>0 Then Begin
            k:=k+1 ;
            IF T[i] <> T[k] Then Begin
                tmp := T[k] ;
                T[k] := T[i] ;
                T[i] := tmp ;
            End;
        End;

```

End ;

```
End ;  
FOR i:=1 To n Do Write(T[i]:3) ;  
End.
```

Exercise 14

```
Program Frequence ;  
Uses WinCRT ;  
Const n=20 ;  
Var T : Array [1..n] of 1..6 ;  
F : Array [1..6] of 0..20 ;  
i : 1..20 ;  
  
Begin  
Randomize ;  
FOR i := 1 To n Do  
Begin  
T[i] := 1 + Random (6) ;  
Write (T[i] : 2) ;  
End ;  
  
FOR i:=1 To 6 Do F[i] := 0 ;  
  
FOR i:=1 To n Do  
F[T[i]] := F[T[i]] + 1 ;  
  
Writeln ;  
FOR i := 1 To 6 Do Write (F[i] : 4) ;  
End.
```

Exercise 15

```
Program Moy_Rang ;  
Uses WinCRT ;  
Const n=30 ;  
Var A, R : Array [1..n] of Real ;  
j, i : Integer ;  
  
Begin  
FOR i:=1 To n Do  
Repeat  
Write ('Note élève ', i, ' : ');  
Readln (A[i]);  
Until (A[i]>=0) and (A[i]<=20);  
  
FOR i:=1 To n Do  
Begin  
R[i]:=1 ;  
FOR j:=1 To n Do  
IF A[i]<A[j] Then R[i]:=R[j]+1 ;  
End ;  
  
Writeln ('Moyens':25, 'Rangs':8);  
FOR i:=1 To n Do  
Writeln (A[i]:25:2, trunc (R[i]):5);  
End.
```

Exercise 16

```
Program Insert_Tab ;  
Uses WinCRT ;  
Const n_max = 100 ;  
Var T : Array [1..n_max] of Char ;  
c : Char ;  
i, k, n : Integer ;  
  
Begin  
Repeat  
Writeln ('Donner un entier '); Readln (n);  
Until (n>=1) and (n<n_max);  
  
Writeln ('Saisir les éléments de T') ;  
FOR i:=1 To n Do Readln (T[i]);  
  
Writeln ('Donner le caractère à insérer');  
Readln (c);
```

```
Repeat  
Writeln ('Donner la position d'insertion');  
Readln (k)  
Until k in [1..n];  
  
{décalage des éléments vers droite}  
FOR i:= n Downto k Do T[i+1] := T[i];  
T[k]:=c;  
FOR i:=1 To n+1 Do Write(T[i]:4);  
End.
```

Exercise 17

```
Program Triangle_Pascal ;  
Uses WinCRT ;  
Type Tab = Array [1..15] of Integer ;  
Var T : Tab ; N : Integer ;  
Procedure init (n : Integer ; Var T : Tab) ;  
Var i : Integer ;  
Begin  
T[1]:=1 ;  
FOR i:=2 To n Do T[i]:=0 ;  
End ;  
Procedure triangle (n : Integer ; Var T : Tab) ;  
Var i, j : Integer ;  
Begin  
Writeln (T[1]);  
FOR i:=2 To n Do  
Begin  
FOR j:=i Downto 2 Do  
Begin  
T[j]:=T[j]+T[j-1];  
Write (T[j], ' ');  
End ;  
Writeln (T[1]);  
End ;  
End ;  
(*****)
```

Exercise 17

```
Program Triangle_Pascal ;  
Uses WinCRT ;  
Type matrice = Array [1..15, 1..15] of Integer ;  
Var T : matrice ; N : Integer ;  
Procedure triangle (n : Integer ; Var T : matrice) ;  
Var l, c : Integer ;  
Begin  
T[1,1]:=1 ;  
FOR l:=2 To n Do  
Begin  
T[l,1]:=1 ;  
FOR c:=2 To l-1 Do  
T[l,c]:=T[l-1,c]+T[l-1,c-1];  
T[l,l]:=1 ;  
End ;  
End ;  
Procedure Afficher (n : Integer ; T : matrice) ;  
Var l, c : Integer ;  
Begin  
FOR l:=1 To n Do  
Begin  
FOR c:=1 To l Do Write (T[l,c], ' ');  
Writeln ;  
End ;  
End ;
```

```

Begin
  Repeat
    Writeln ('Donner la taille du triangle : '); Readln (n);
    Until n in [2..15];
    Init (n, T);
    Triangle (n, T);
  End.

```

```

(*****)
Begin
  Repeat
    Writeln ('Donner la taille du triangle : '); Readln (n);
    Until n in [2..15];
    Triangle (n, T);
    Afficher (n, T);
  End.

```

Exercice 18

```

Program transpose_matrice;
uses wincrt;
const nmax=10;
type Mat=array[1..nmax,1..nmax] of integer;
var M:Mat;
    n:integer;
Procedure saisie (Var n:integer);
begin
  repeat
    writeln('Donner n :');
    readln(n);
  until n in [1..nmax];
end;
Procedure remplir (Var M:Mat; n:integer);
var i, j:integer;
begin
  For i:=1 to n do
    For j:=1 to n do
      begin
        Writeln('Donner M[',i,',',j,']');
        readln(M[i,j]);
      end;
end;
end;

```

```

Procedure Transpose (Var M:Mat;n:integer);
var a ux,i,j:integer;
begin
  For i:=1 to n do
    For j:=1 to i-1 do
      begin
        aux:=M[i,j];
        M[i,j]:=M[j,i];
        M[j,i]:=aux;
      end;
end;
Procedure Affiche ( M:Mat; n:integer);
var i, j:integer;
begin
  For i:=1 to n do
    begin
      For j:=1 to n do Write( M[i,j], ' ');
      writeln;
    end;
end;
{Programme Principal}
begin
  saisie (n);
  remplir (M, n);
  Transpose (M, n);
  Affiche (M, n);
end.

```

Exercice 19

```

Program TRI_SELECTION ;
Uses Wincrt ;
Const n = 20 ;
Type Tab = Array [1.. n] of String ;
Var T : Tab ;
    i , j , posmin : Integer ;
    tmp : String;
Begin
  Writeln ('Remplir le tableau par des chaînes :');
  FOR i := 1 TO n DO Readln (T[i]) ;

  FOR i := 1 TO n-1 DO
    Begin
      posmin := i ;
      FOR j := i+1 TO n DO
        IF T[j] < T[posmin] Then posmin := j ;
      IF i <> posmin Then Begin
        tmp := T[i] ;
        T[i] := T[posmin] ;
        T[posmin] := tmp ;
      End ;
    End ;

  Writeln ('Tableau trié :');
  FOR i := 1 TO n DO Writeln (T[i]) ;
End.

```

Exercice 20

```

Program Moy_Rang;
Uses Wincrt;
Const n=30;
Var Nom : Array [1..n, 1..2] of String [50];
    Note : Array [1..n, 1..4] of Real;
    j, i, a, b : Integer;
Begin
  FOR i:=1 To n Do
    Begin
      Writeln ('Elèves n° ', i);
      Write ('Nom : '); Readln (nom[i,1]);
      Write ('Prénom : '); Readln (nom[i,2]);
      FOR j:=1 To 2 Do
        Repeat
          Write ('Note ', j, ' : ');
          Readln (note[i,j]);
        Until (note[i,j]>=0) and (note[i,j]<=20);
        note[i,3] := (note[i,1]+note[i,2]*2)/3;
      End;
    End;
  FOR i:=1 To n Do
    Begin
      note[i,4]:=1;
      FOR j:=1 To n Do
        IF note[i,3]<note[j,3] Then note[i,4]:=note[i,4]+1;
      End;
      Writeln ('Nom', 'Prénom':15, 'Note 1':10, 'Note 2':10,
        'Moyens':10, 'Rangs':10);
    End;
  FOR i:=1 To n Do
    Begin
      a:=Length (nom[i,1]);
      b:=Length (nom[i,2]);
      Writeln (nom[i,1], nom[i,2]:(12-a+b),
        note[i,1]:(15-b):2, note[i,2]:10:2,
        note[i,3]:10:2, trunc(note[i,4]):9);
    End;
  End.

```

Exercice 21

```

(*****)

```

```

Program Intersection_Tab;
Uses Wincrt;
Type vect = Array [1..99] of Integer;
Var T1, T2, inter : vect;
    n, m : Integer;
(*****)
Procedure saisie_int (Var nf : Integer);
Begin
    Repeat
        Write ('N = ');
        Readln (nf);
    Until nf in [3..99];
End;
(*****)
Procedure remplir_tab (nf : Integer ; Var A:vect);
Var i, j : Integer;
Begin
    Writeln ('Saisir la case 1');
    Readln (a[1]);
    FOR i:=2 To nf Do
        Repeat
            Writeln ('Saisir la case ', i);
            Readln (A[i]);
            j:=i-1;
            While (j>1) and (A[i] <> A[j]) Do j:=j-1;
        Until A[i]<>A[j];
    End;
End;

```

```

Procedure intersection (nf : Integer ; A1, A2:vect;
    Var p : Integer ; Var B:vect);
Var i, j : Integer;
Begin
    p:=0;
    FOR i:=1 To nf Do
        Begin
            j:=0;
            Repeat
                j:=j+1;
            Until (j=nf) Or (A1[i]=A2[j]);
            IF A1[i]=A2[j]
                Then Begin
                    p:=p+1;
                    B[p]:=A1[i];
                End;
            End;
        End;
    End;
(*****)
Procedure affiche_tab (nf : Integer ; A:vect);
Var i : Integer;
Begin
    FOR i:=1 To nf Do Write (A[i], ' ');
End;
(***** p.p *****)
Begin
    saisie_int (n); remplir_tab (n, T1); remplir_tab( n, T2);
    intersection (n, T1, T2, m, inter);
    affiche_tab (n, T1); Writeln ; affiche_tab (n, T2);
    Writeln ;
    affiche_tab (m, inter);
End.

```

Exercise 22

```

program tri_2_criteres;
uses wincrt,ecran;
const n=10;
type tab=array[1..n] of string;
var t:tab;
    i,j,pos:integer;
    aux:string;
begin
    writeln('Remplir T :');
    for i:=1 to n do
        repeat
            write('ch = ');
            readln(t[i]);
            until t[i]<>'';
        (*****)
        for i:=1 to n-1 do
            begin
                pos:=i;
                for j:=i+1 to n do
                    if (length(t[j])<length(t[pos])) OR
                        ((length(t[j])=length(t[pos]))AND(t[j]<t[pos]))
                        then pos:=j ;
                if i<>pos then begin
                    aux:=t[i] ;
                    t[i]:=t[pos];
                    t[pos]:=aux;
                    end;
            end;
        (*****)
        for i:=1 to n do writeln (t[i]);
    end.

```

Exercise 24

```

Program tri_2_criteres;
uses wincrt,ecran;
type tch=array[1..10] of string[20]; tc=array[1..10] of char;
var n:integer; t:tch; c:tc;
(*****)
procedure saisie(var n:integer;var t:tch;var c:tc);
var i:integer;
begin
    write ('N = ');readln(n);
    writeln ('remplir les tableaux T et C :');
    for i:=1 to n do
        begin
            write('nom = '); readln(t[i]);
            repeat write ('couleur = '); readln (c[i]); until c[i] in ['B','N'];
            end;
        end;
    (*****)
procedure tri(n:integer;var t:tch;var c:tc);
var i:integer; permut:boolean; aux:string; tmp:char;
begin
    repeat
        permut:=false;
        for i:=1 to n-1 do
            if (c[i]>c[i+1])or((c[i]=c[i+1])and(t[i]>t[i+1]))
                then begin
                    aux:=t[i] ; t[i]:=t[i+1]; t[i+1]:=aux;
                    tmp:=c[i] ; c[i]:=c[i+1]; c[i+1]:=tmp;
                    permut:=true
                end;
        n:=n-1
        until (permut=false) or (n=1);
    end;
    (*****)
procedure affiche (n:integer;t:tch;c:tc);
var i:integer;
begin
    for i:=1 to n do writeln(t[i],',',c[i]);
end;
(***** p.p *****)
Begin
    Saisie (n,t,c); tri (n,t,c); affiche (n,t,c);

```

End.

Exercice 23

```
Program tri_bulles_bidirectionnel;
uses wincrt,ecran;
type tab=array[1..25] of integer;
var t:tab;
    n:integer;
(*****)
procedure saisir(var n:integer);
begin
  repeat
    writeln('Donner un entier entre 5 et 25');
    readln(n);
  until n in [5..25];
end;
(*****)
procedure remplir (var t:tab ; n:integer);
var i:integer;
begin
  randomize;
  for i:=1 to n do T[i]:=1+random(100);
end;
(*****)
procedure trier (var T:tab ;n:integer);
var i,aux,debut,fin:integer;
    permut:boolean;
(*****)
begin
  debut:=1;fin:=n;
  repeat
    permut:=false;

    for i:=debut to fin-1 do
      if t[i]>t[i+1]
      then begin
        aux:=T[i];
        T[i]:=T[i+1];
        T[i+1]:=aux;
        permut:=true;
      end;
    fin:=fin-1;

    for i:=fin downto debut+1 do
      if t[i]<t[i-1]
      then begin
        aux:=T[i];
        T[i]:=T[i-1];
        T[i-1]:=aux;
        permut:=true;
      end;

    debut:=debut+1;
  until (permut=false) or (debut>=fin);
end;
(*****)
procedure afficher(T:tab ; n:integer);
var i:integer;
begin
  for i:=1 to n do write(T[i],' ');
end;
(***** p.p. *****)
BEGIN
  saisir(n);
  remplir(t,n);
  writeln('Tableau avant le tri :');
  afficher(t,n);
  trier(t,n);
  writeln;
  writeln('Tableau après le tri :');
  afficher(t,n);
END.
```

Exercice 25

```
program fusion;
uses wincrt,ecran;
type tab=array [1..20] of integer;
var v1,v2,v3:tab;
    n,m,c:integer;
(*****)
procedure lecture (var taille:integer);
begin
  repeat
    readln(taille);
  until taille in [2..20];
end;
(*****)
procedure remplir(var t:tab; taille:integer);
var i:integer;
begin
  for i:= 1 to taille do readln(t[i]);
end;
(*****)
procedure trier (taille:integer;var t:tab);
var i,tmp, min,j:integer;
begin
  for i:=1 to taille-1 do
    begin
      min:=i;
      for j:=i+1 to taille do
        if t[j]<t[min] then min:=j;
      if i<>min then begin
        tmp:=t[i];
        t[i]:=t[min];
        t[min]:=tmp;
      end;
    end;
end;
(*****)
procedure fusionner(v1,v2:tab;var v3:tab;n,m:integer;var c:integer);
var i,c1,c2:integer;
begin
  c1:=1; c2:=1; c:=0;
  repeat
    c:=c+1;
    if v1[c1]<v2[c2]
    then begin
      v3[c]:=v1[c1];
      c1:=c1+1;
    end
    else begin
      v3[c]:=v2[c2];
      c2:=c2+1;
    end
  until (c1>n) or (c2>m);
  if c1>n then
    for i:=c2 to m do
      begin
        c:=c+1;
        v3[c]:=v2[i];
      end
  else
    for i:=c1 to n do
      begin
        c:=c+1;
        v3[c]:=v1[i];
      end;
  end;
(*****)
procedure afficher(t:tab; taille:integer);
var i:integer;
begin
  writeln('Tableau fusion :');
  for i:= 1 to taille do
    write(t[i]:4);
  end;
end;
```



```

(*****p.p*****)
begin
  write('Taille V1 : ');lecture(n);
  write('Taille V2 : ');lecture(m);
  writeln('Remplir V1 : ');remplir(v1,n);
  writeln('Remplir V2 : ');remplir(v2,m);
  trier(n,v1);
  trier(m,v2);
  fusionner(v1,v2,v3,n,m,c);
  afficher(v3,c);
end.

```

Exercice 26

```

Program temps_tris;
uses wincrt,windos,ecran;
type tab=array[1..1000] of real;
var t,t1,t2:tab;
    n:integer;
    hi1,hi2,mi1,mi2,si1,si2,csi1,csi2,hs1,hs2,
    ms1,ms2,ss1,ss2,css1,css2,ts1,ti1:word;
(***** lecture et duplication *****)
procedure lecture_duplic(var n:integer;var t,t1,t2:tab);
var i:integer;
begin
  Writeln('Saisir un entier pour la taille des tableaux');
  Readln(n);
  randomize;
  for i:=1 to n do
    begin
      t[i]:=100*random; { réel aléatoire entre ]0..100[ }
      t1[i]:=t[i];
      t2[i]:=t[i];
    end;
end;
(***** TRI SELECTION *****)
Procedure tri1 (n:integer;var t1:tab);
var pm,i:integer;
(*****)
Function posmin(d,f:integer;t:tab):integer;
var i,pmin,j:integer;
begin
  pmin:=d;
  for j:=d+1 to f do
    if t[j] < t[pmin] then pmin:=j;
  posmin:=pmin;
end;
(*****)
Procedure permut (var x,y:real);
var aux:real;
begin
  aux:=x;
  x:=y;
  y:=aux;
end;
(*****)
begin
  for i:=1 to n-1 do
    begin
      pm:=posmin(i,n,t1);
      if pm<>i then permut(t1[pm],t1[i]);
    end;
end;

```

Exercice 27

```

Procedure Trier (n:integer ; T:tab;var rang,s:tab);
Var i,j : integer;
BEGIN
  FOR i:=1 TO n DO s[i]:=1;

  FOR i:=1 TO n-1 DO
    FOR j:=i+1 TO n DO
      IF T[i]>T[j]
        THEN s[i]:=s[i]+1
        ELSE s[j]:=s[j]+1;

```

***** TRI INSERTION *****)

```

procedure tri2 (n:integer;var t2:tab);
var j,i:integer;
    tmp:real;
(*****)
procedure decaler (var t2:tab;var j:integer;i:integer);
begin
  j:=i;
  WHILE (j>1)and(t2[j-1]>tmp) DO
    Begin
      t2[j]:=t2[j-1];
      j:=j-1;
    End ;
end;
(*****)
Begin
  for i:=2 to n do
    if t2[i]<t2[i-1]
      then Begin
        tmp:=t2[i];
        Decaler(t2,j,i);
        t2[j]:=tmp;
      End ;
  End;
(***** Affichage *****)
procedure affiche(n:integer;t:tab);
var i:integer;
begin
  for i:=1 to n do write(t[i]:2:2, ' ');
end;
(***** Programme principal *****)
BEGIN
  lecture_duplic(n,t,t1,t2);
  gettime(hs1,ms1,ss1,css1);
  tri1(n,t1);
  gettime(hs2,ms2,ss2,css2);
  ts1:=(hs2-hs1)*3600*100+(ms2-ms1)*60*100+(ss2-s1)*100+css2-
  css1;
  gettime(hi1,mi1,si1,csi1);
  tri2(n,t2);
  gettime(hi2,mi2,si2,csi2);
  ti1:=(hi2-hi1)*3600*100+(mi2-mi1)*60*100+(si2-si1)*100+csi2-
  csi1;
  affiche(n,t1); readln;
  affiche(n,t2); readln;
  writeln('tri selection : ',ts1, ' Centième de seconde');
  writeln('tri insertion : ',ti1, ' Centième de seconde');
END.

```

Exercice 28

```

program long_suite;
uses wincrt, ecran;
const n=20;
type tab=array[1..n] of char;
var t:tab; max,suite:string; i:integer;
begin
  for i:=1 to n do readln(t[i]);
  max:=t[1];
  suite:=t[1];
  for i:=2 to n do

```

```
FOR i:=1 TO n DO rang[s[i]]:=i;
END;
```

```
if t[i]=t[i-1]
then suite:=suite+t[i]
else begin
if length(suite)>length(max) then max:=suite;
suite:=t[i];
end;

writeln (max[1], length(max));
end.
```

Exercice 29

```
Program symetrique;
uses wincrt;
type tab=array [1..200] of integer;
var t:tab;
n,i:integer;
(*****)
Procedure saisie(var n:integer;var t:tab);
var i:integer;
begin
repeat
write('N = ');
readln(n);
until n in [5..200];
Randomize;
for i:=1 to n do
T[i]:=100+Random(900);
end;
(*****)
Function verif (x:integer):boolean;
var ch:string;
begin
str(x,ch);
verif:= ch[1] = ch[3]
end;
(***** p.p *****)
BEGIN
saisie(n,t);
writeln('les nombres symétriques de T sont: ');
for i:= 1 to n do
if verif(t[i]) then write(t[i]:4);
END.
```

Exercice 30

```
Program element_manquant ;
uses wincrt;
type tab=array [1..20] of integer;
var t:tab;
n:integer;
(*****)
Procedure saisie(var n:integer;var t:tab);
var i:integer;
begin
repeat
writeln('Donner le nombre d"éléments N, 2<=n<=20');
readln(n);
until n in [2..20];
repeat
write('T[1] : '); readln(T[1]);
until T[1]>=0;
for i:=2 to n do
repeat
write('T['+i+'] : ');
readln(T[i]);
until T[i]>=T[i-1];
end;
(*****)
Procedure manque (n:integer;t:tab);
var x,i,j:integer;
begin
write('Les entiers manquants sont : ');
x:=0;
for i:=2 to n do
if (T[i]<>T[i-1]+1)
then for j:=(T[i-1]+1) to (T[i]-1) do
begin
write(j,' ');
x:=x+1;
end;
write('; leur nombre est : ',x);
end;
(***** p.p *****)
begin
saisie(n,t);
manque(n,t);
end.
```

Exercice 31

```
Program El_frequent;
Uses Wincrt ;
Type tab1=Array [1..20] of 0..9 ;
tab2=Array [0..9] of 0..20 ;
Var T:tab1;
F:tab2;
n:integer;
(*****)
Procedure Saisir (var n:integer);
begin
Repeat
writeln('Saisir un entier N, (5<=n<=20)');
readln(n);
Until n in [5..20];
end;
```

```
Procedure affiche(n:integer;t:tab1;var f:tab2);
var i,max:integer;
Begin
For i:=0 To 9 Do F[i] := 0 ;

For i:=1 To n Do
F[T[i]] := F[T[i]] + 1 ;

max:=1;
For i:= 2 To 9 Do
if F[i]>F[max] then max:=i;

writeln(max,', son nombre d"occurrence est ', F[max]);
End;
(***** p.p *****)
```

```
(*****)  
Procedure remplir (n:integer;var t:tab1);  
var i:integer;  
begin  
  randomize;  
  for i:=1 to n do  
    begin  
      t[i]:=random(10);  
      write(t[i]:3);  
    end;  
  writeln;  
end;  
(*****)
```

```
BEGIN  
  saisir(n);  
  remplir(n,t);  
  affiche(n,t,f);  
END.
```

fenni salah

LES CHAINES DE CARACTERES

Exercice 2

```

Program Palindrome;
Uses WinCRT;
Var ch, inv : String;
    i : Integer;
Begin
  Writeln ('Saisir une chaîne'); Readln (ch);
  inv := '';
  FOR i := Length (ch) Downto 1 Do
    inv := inv + ch[i];
  IF ch = inv
  Then Writeln (ch, ' est palindrome')
  Else Writeln (ch, ' n'est pas palindrome');
End.
  
```

Exercice 4

```

Program Chaîne_Majus_Minus;
Uses WinCRT;
Var ch : String;
    i : Integer;
Begin
  Writeln ('Saisir une chaîne de caractères'); Readln (ch);
  FOR i:=1 To Length (ch) Do
    IF ch[i] in ['a'..'z']
    Then ch[i]:=ch[i]
    Else ch[i]:=CHR (ORD (ch[i]) + 32);
  Writeln (ch);
  Writeln ;
  FOR i:=1 To Length (ch) Do ch[i]:=upcase (ch[i]);
  Writeln (ch);
End.
  
```

Exercice 5

```

Program Chaîne1 ;
Uses WinCRT ;
Var i : Integer ; ch: String ;
    test : Boolean;
Begin
  Repeat
    Writeln ('Donner un mot en majuscule') ; Readln (ch) ;
    test:=True;
    i:=0 ;
    Repeat
      i:=i+1;
      IF Not(ch[i] in ['A'..'Z']) Then test:=False;
    Until (test=False) Or (i=Length (ch));
  Until test=True ;
  FOR i := 1 To Length (ch) Do
    Writeln (COPY (ch, 1, i));
End.
  
```

Exercice 7

```

Program Renverser_ch ;
Uses WinCRT ;
Var p : Integer ;
    chr, chd : String ;
Begin
  Writeln ('Saisir une phrase') ; Readln (chd) ;
  chr := '' ;
  p := POS(' ', chd) ;
  While p <> 0 Do
    Begin
      chr := '' + COPY (chd, 1, p-1) + chr ;
      DELETE (chd, 1, p) ;
      p := POS (' ', chd) ;
    End ;
  chr := chd + chr ;
  Writeln ('Phrase renversée est : ', chr) ;
End.
  
```

Exercice 3

```

Program chaîne_inverse;
Uses winCRT;
Var ch : string;
  (*****)
Function miroir (ch:string):string;
var i, l : integer; c : char;
begin
  l:=length(ch);
  for i:=1 to l div 2 do
    begin
      c:=ch[i];
      ch[i]:=ch[l-i+1];
      ch[l-i+1]:=c;
    end;
  miroir:=ch;
end;
  (***** p.p *****)
BEGIN
  write('ch = ');readln(ch);
  writeln('l'inverse de ',ch,' est : ', miroir(ch));
END.
  ===== solution 2 =====
Function miroir (ch:string) : string;
var i, l : integer; mirch : string;
begin
  l:=length(ch);
  mirch:=ch;
  for i:=1 to l do mirch[i] := ch[l-i+1];
  miroir:=mirch;
end;
  
```

Exercice 5

```

Program Chaîne2;
Uses WinCRT;
Var ch : String; i, lg : Integer;
    test : Boolean;
Begin
  Repeat
    Writeln ('Donner un mot en majuscule') ; Readln (ch) ;
    test:=True; i:=0 ;
    Repeat
      i:=i+1;
      IF Not (ch[i] in ['A'..'Z']) Then test:=False;
    Until (test=False) Or (i=Length (ch));
  Until test=True ;
  lg:=Length (ch);
  FOR i:=1 To lg Do
    Writeln (COPY (ch, 1, i), COPY (ch, lg-i+1, i));
End.
  
```

Exercice 6

```

Program Espace_superflus ;
Uses WinCRT ;
Var ch : String;
    i, p : Integer ;
Begin
  Writeln ('Donner une chaîne'); Readln (ch);
  Repeat
    P := POS (' ', ch); {position de 2 espaces dans ch}
    IF p<>0 Then DELETE (ch, p, 1);
  Until p=0 ;
  IF ch[1]=' '
  Then DELETE (ch, 1, 1);
  IF ch[Length(ch)]=' '
  Then DELETE (ch, Length (ch), 1);
  Writeln ('La chaîne devient : ', ch);
End.
  
```

Exercice 8

```

Program Occurrence_car;
Uses WinCRT;
Var ch : String;
    i, j, n : Integer;
Begin
  Writeln ('Saisir une chaîne'); Readln (ch);
  FOR i:=1 To Length (ch) Do
  Begin
    n:=0;
    FOR j:=1 To Length (ch) Do
      IF ch[i]=ch[j]
        Then n:=n+1;
    IF i = POS (ch[i], ch)
      Then Writeln ('Occurrence de ', ch[i], ' = ', n);
  End;
End.

```

Exercice 10

```

Program Sans_Redondance ;
Uses WinCRT;
Var ch1, ch2, ch3, aux : String; i : integer;
Begin
  Write('Chaîne 1 = ');Readln (ch1);
  Write('Chaîne 2 = ');Readln (ch2);
  if length(ch1)>length(ch2)
  then begin
    aux:=ch1;
    ch1:=ch2;
    ch2:=aux;
  end;
  ch3:="";
  FOR i:=1To Length(ch1) Do
    IF (POS(ch1[i],ch2)<>0)and(POS(ch1[i],ch3)=0)
      Then ch3:=ch3+ch1[i];
  Writeln(ch3);
End.

```

Exercice 12

```

Program Anagrammes;
uses winCRT;
var mot1, mot2 : string;
(*****)
procedure saisie_ch (var m1, m2 : string);
begin
  repeat
    writeln ('donner deux mots : ');
    readln (m1);
    readln (m2);
  until (m1 > "") and (m2 > "");
end;
(*****)
function trie (mot : string) : string;
var i, j, n : integer;
procedure permut (var a, b : char);
var aux : char;
begin
  aux:=a; a:=b; b:=aux;
end;
begin
  n:=length (mot);
  for i:=1 to n-1 do
    for j:=i+1 to n do
      if mot[i]>mot[j]
        then permut (mot[i], mot[j]);
    trie := mot;
end;
(*****)
begin
  saisie_ch (mot1, mot2);

```

Exercice 9

```

Program Occurrence_mot ;
uses winCRT;
var ch, mot : string; nb, i, k: integer;
begin
  repeat
    writeln ('saisir un texte'); readln (ch);
  until length (ch)>20;
  writeln ('saisir un mot'); readln (mot);
  k:=length (mot); nb:=0; i:=1;
  repeat
    if (ch[i] = mot[1]) and (mot = copy (ch, i, k))
      then begin
        nb:=nb+1;
        i:=i+k;
      end
    else i:=i+1;
  until i>length(ch);
  writeln (mot, ' figure dans le texte ', nb, ' fois');
end.

```

Exercice 11

```

Program Aerer_ch ;
Uses WinCRT ;
Var k : Byte ;
    ch : String ;
Begin
  Writeln ('Saisir une chaîne') ;
  Readln (ch) ;

  k := 0 ;
  repeat
    k:=k+2;
    Insert (' ', ch, k) ;
  Until k = length(ch)-1;

  Writeln ('Chaîne aérée = ', ch) ;
End.

```

Exercice 12

```

Program anagrammes;
uses winCRT;
var mot1, mot2:string;
(*****)
procedure saisie_ch(var m1,m2:string);
begin
  repeat
    writeln ('donner deux mots : ');
    readln (m1);
    readln(m2);
  until (m1 > "") and (m2 > "");
end;
(*****)
function anagram (mot1,mot2:string):boolean;
var p:integer;
begin
  anagram:=false;
  repeat
    p:=pos(mot1[1],mot2);
    if p>0
      then begin
        delete(mot1,1,1);
        delete(mot2,p,1);
      end;
    until (p=0) or (mot1="");
    if (mot1="") and (mot2="") then anagram:=true;
end;
(*****)
begin
  saisie_ch (mot1, mot2);
  if anagram (mot1, mot2)

```

```

if trie (mot1) = trie (mot2)
then writeln (mot2, ' est une anagramme de ', mot1)
else writeln (mot2, ' n'est pas une anagramme de ', mot1);
end.

```

Exercice 13

```

Program Pos_ch;
Uses Wincrt;
Var ch1, ch2 : String;
    i, p : Integer;
Begin
  Write ('ch1 = '); Readln (ch1);
  Write ('ch2 = '); Readln (ch2);
  i:=1 ;
  p:=0;
  Repeat
    IF ch1 = COPY (ch2, i, Length (ch1)) Then p:=i;
    i:=i+1;
  Until (Length (ch2)-i < Length (ch1)) Or (p<>0) ;
  Writeln ('La chaîne ', ch1, ' occupe la position ', p,
    ' dans la chaîne ', ch2);
End.

```

Exercice 15

```

Program Jeu_pendu ;
Uses Wincrt;
type chain=String [50];
Var se, ma:chain;
    rep, let : Char;
(*****)
Function controle_saisie (se:chain) : Boolean;
Var i : Integer;
    r : Boolean;
Begin
  r:=True; i:=0 ;
  Repeat
    i:=i+1;
    IF Not (se[i] in ['A'..'Z']) Then r:=False;
  Until (r=False) Or (i=Length (se));
  controle_saisie := r;
End;
(*****)
Procedure masquer (se:chain; Var ma : chain);
Var i : Integer;
Begin
  ma:=se;
  FOR i:= 2 To (Length (se)-1) Do ma[i]:='-';
End;
(*****)
Procedure saisie_let (Var let : Char);
Begin
  Writeln ('Donner une lettre ');
  Readln (let);
  let:=upcase (let);
End;
(*****)
Procedure dévoiler (se : chain; let : Char; Var ma : chain);
Var i : Integer; r : Boolean;
Begin
  r:=False;
  FOR i:=2 To (Length (se)-1) Do
    IF se[i]=let
      Then Begin
        ma[i]:=let;
        r:=True;
      End;
  IF r=False Then Writeln ('Echec');
End;
(*****)
Procedure partie_jeu (se, ma : chain ; let : Char);
Var nb : Integer;
Begin
  ClrScr;
  Writeln (ma);
  nb:=0;
  Repeat

```

```

then writeln (mot1, ' est une anagramme de ', mot2)
else writeln (mot1, ' n'est pas une anagramme de ', mot2);
end.

```

Exercice 14

```

Program Copie_ch;
Uses Wincrt;
Var ch1, chr : String;
    i, p, n, l : Integer;
Begin
  Write ('ch1 = '); Readln (ch1);
  l:=Length (ch1);
  Repeat
    Write ('Position = '); Readln (p);
    Write ('Nbre caractères = '); Readln (n);
  Until (p in [1..l]) and (n in [1..l]) and (p+n<=l+1);
  chr:="";
  FOR i:=p To p+n-1 Do
    chr:=chr+ch1 [i];
  Writeln ('La chaîne copiée est : ', chr);
End.

```

Exercice 16

```

Program Chaines_inverses;
Uses Wincrt;
Type Tab=Array [1..100] of String [50];
Var p, q : Tab;
    i, n : Integer;
(*****)
Procedure saisie_entier (Var m : Integer);
Begin
  Repeat
    Writeln ('Donner un entier'); Readln (m);
  Until (1<m) and (m<100);
End;
(*****)
Procedure saisie_tab (m : Integer; Var T : Tab);
Var i, j : Integer; test : Boolean;
Begin
  FOR i:=1 To m Do
    Repeat
      Writeln ('Donner l'élément d'ordre ', i); Readln (T[i]);
      j:=0; test:=True;
      While (test=True) and (j<Length (T[i])) Do
        Begin
          j:=j+1;
          IF Not (T [i, j] in ['0'..'9']) Then test:=False;
        End;
      Until (test=True) and (T[i] <>'');
    End;
End;
(*****)
Function inverse (ch : String) : String;
Var i : Integer; chinvs : String;
Begin
  chinvs:="";
  FOR i:=Length (ch) Downto 1 Do
    chinvs:=chinvs+ch[i];
  inverse:=chinvs;
End;
(*****)
Procedure Affiche_tab (m : Integer; T : Tab);
Var i : Integer;
Begin
  FOR i:=1 To m Do Write (T[i], ' ');
End;
(***** p.p *****)
Begin
  saisie_entier (n);
  saisie_tab (n, p);
  FOR i:=1 To n Do
    q[i] := inverse (p[i]);
  affiche_tab (n, q);
End.

```



```

nb:=nb+1;
saisie_let (let);
devoiler (se, let, ma);
Writeln (ma);
Until (nb=length (se)) Or (ma=se);
IF ma=se
Then Writeln ('Bravo, trouvé en ', nb, ' propositions')
Else Writeln ('Perdu le mot à deviner est ', se);
End;
(***** p, p *****)
Begin
Writeln ('Le jeu du pendu'); Writeln ;
Repeat
Repeat
Write ('Rentrez un mot secret en majuscule : ');
Readln (se);
Until controle_saisie (se);
masquer (se, ma);
partie_jeu (se, ma, let);
Write ('Voulez-vous rejouer (o/n) ? ');
Readln (rep);
Until Ucase (rep) = 'N';
End.

```

Exercice 17

```

Program conversion_base1_base2;
uses wincrt, ecran;
var b1, b2:integer;
nch:string;
(*****)
Procedure saisie_base (var b1,b2:integer);
begin
repeat
write('Base b1 = ');readln(b1);
write('Base b2 = ');readln(b2);
until (b1 in [2..16]) and (b2 in [2..16]) and (b1<>b2);
end;
(*****)
Procedure saisie_nombre (var nch:string; b1:integer);
Const chb='0123456789ABCDEF';
Var i:integer;
test:boolean;
begin
repeat
writeln ('Donner un nombre en base ', b1);
readln(nch);
test:=true;
for i:=1 to length(nch) do
if (pos(nch[i],chb)>b1) or (pos(nch[i],chb)=0)
then test:=false;
until test=true;
end;

```

```

(**** Conversion de la base b1 vers base 10 *****)
Function Conv_b1_10 (nch:string; b1:integer) : longint;
var err,i,n:integer;
dec,puiss:longint;
begin
dec:=0;puiss:=1;
for i:=length(nch) downto 1 do
begin
if nch[i] in ['0'..'9']
then Val(nch[i], n, err)
else n:=ord(nch[i])-55;
dec:=dec+n*puiss;
puiss:=puiss*b1;
end;
conv_b1_10:=dec;
end;
(**** Conversion de la base 10 vers la base b2 *****)
Function Conv_10_b2 (nd:longint; b2:integer) : string;
var ch1, chb2:string;
r:integer;
begin
chb2:="";
repeat
r:= nd mod b2;
if r in [0..9] then str(r,ch1)
else ch1:= chr(55+r);
chb2:=ch1+chb2; (*insert (ch1,chb2,1)*)
nd:= nd div b2 ;
until (nd = 0);
conv_10_b2:=chb2;
end;
(***** p,p *****)
Begin
saisie_base(b1,b2);
saisie_nombre(nch,b1);
writeln('(',nch,')',b1,' = (', conv_10_b2 (conv_b1_10 (nch, b1), b2),
'), b2);
End.

```

Exercice 18

```

Program Nbre_Rond;
Uses Wincrt;
Var reste, c : String;
n, m, i, j, n1, n0 : Integer;
Begin
FOR i:=1 To 1000 Do
Begin
n:=i;
m:=i;
reste:="";
Repeat

```

```

n1:=0; n0:=0;
FOR j :=1 To Length (reste) Do
Begin
IF reste[j]='1' Then n1:=n1+1;
IF reste[j]='0' Then n0:=n0+1;
End;
IF n1=n0 Then Writeln (m, ' est ROND')
Else Writeln (m, ' n'est pas ROND');
Readln ;
End;
End.

```

```

STR (n mod 2, c);
reste:= c+reste;
n:=n div 2;
Until n=0;

```

Exercise 20

```

Program sablier;
uses wincrt;
var esp,ch:string;
(*****)
Procedure affiche_bas(var esp:string;ch:string);
var ch2:string; n,i:integer;
begin
  esp:="";n:=0;i:=1;
  writeln(ch);
  repeat
    esp:=' '+esp;
    n:=n+2;i:=i+1;
    ch2:=esp+copy(ch, i, length(ch)-n);
    writeln(ch2);
  until length(copy(ch, i, length(ch)-n))=1;
end;
(*****)
Procedure affiche_haut(esp,ch:string);
Var i,n:integer; ch2:string;
begin
  i:=-1;n:=1;
  repeat
    n:=n+2;i:=i+1;
    delete(esp,1,1);
    ch2:=esp+copy(ch, length(ch) div 2-i, n);
    writeln(ch2);
  until ch2=ch;
end;
(*****pp*****)
begin
  repeat
    write('CH = '); readln(ch);
  until (ch<>'') and (odd(length(ch)));
  affiche_bas(esp,ch);
  affiche_haut(esp,ch);
end.

```

Exercise 21

```

Program Totalogram;
uses wincrt;
var ch:string;
(*****)
Function lettre_maj_esp (ch:string):boolean;
var i:integer;
    test:boolean;
begin
  i:=1; test:=true;
  while (i<=length(ch)) and (test) do
    if ch[i] in ['A'..'Z', ' ']
      then i:=i+1
      else test:=false;
  lettre_maj_esp:=test;
end;
(*****)
Function totalogramme (ch:string):boolean;
var p:integer; test:boolean;
begin
  ch:=ch+' ';
  repeat
    p:=pos(' ',ch);
    test:=ch[1]=ch[p-1];
    delete(ch,1,p);
  until (test=false) or (ch="");
  totalogramme:=test;
end;
(***** p,p *****)

```

Exercise 19

```

Program suite_mystere;
uses wincrt, ecran;
var ligne, lignsuiv, c : string;
    n, l, nb, j, i : integer;
begin
  write ('N = '); readln (n);
  ligne:='1';
  for i:=1 to n do
    begin
      writeln (ligne);
      l:=length (ligne);
      nb:=1;
      lignsuiv:="";
      for j:=1 to l do
        if ligne[j] = ligne[j+1]
          then nb:=nb+1
          else begin
            str (nb, c);
            lignsuiv:= lignsuiv + c + ligne[j];
            nb:=1;
          end;
        ligne:= lignsuiv;
      end;
    end.

```

Exercise 22

```

Program ch_distincte;
uses wincrt;
var ch:string;
(*****)
procedure saisie (var ch:string);
begin
  repeat
    writeln('Saisir une chaîne non vide :');
    readln(ch);
  until ch<>'';
end;
(*****)
Function distincte (ch:string):boolean;
var i:integer;
    test:boolean;
begin
  i:=0;
  test:=true;
  repeat
    i:=i+1;
    if pos(ch[i],ch)<>i then test:=false;
  until (test=false) or (i=length(ch));
  distincte:=test;
end;
(***** p,p *****)
begin
  saisie(ch);
  if distincte(ch)

```

```

begin
repeat
  writeln('Saisir une chaîne en majuscule :');
  readln(ch);
until lettre_maj_esp(ch);
if totalogramme(ch)
  then writeln('totalogramme')
  else writeln('non totalogramme');
end.

```

```

then writeln('cette chaîne est distincte')
else writeln('cette chaîne est non distincte');
end.

```

Exercice 23

```

Program Exercice_23;
uses wincrt;
var ch:string;
(*****)
Procedure saisie(var ch:string);
var i:integer;
begin
  repeat
    writeln('saisir une chaine alphanétique :');
    readln(ch);
    i:=1;
    while upcase(ch[i]) in ['A'..'Z'] do i:=i+1;
  until (i>length(ch)) and (length(ch) in [1..50]);
end;
(*****)
Function construire (ch:string):string;
var i, p1, p2, p3 : integer;
    res:string;
begin
  res:="";
  p1:=1; p2:=1; p3:=1;
  for i:=1 to length(ch) do
    case ch[i] of
      'A'..'Z' : if not (ch[i] in ['A','E','I','O','U','Y'])
        then begin
            insert(ch[i],res,p1);
            p1:=p1+1;
            p2:=p2+1;
            p3:=p3+1;
          end
        else begin
            insert(ch[i],res,p2);
            p2:=p2+1;
            p3:=p3+1;
          end;
      'a'..'z' : if not (ch[i] in ['a','e','i','o','u','y'])
        then begin
            insert(ch[i],res,p3);
            p3:=p3+1;
          end
      else res:=res+ch[i];
    end; {fin selon}
  construire:=res;
end;
(***** p.p *****)
BEGIN
  saisie(ch);
  writeln('La chaîne devient : ',construire(ch));
END.

```

Exercice 24

```

Program long_palindrome;
uses wincrt;
var ch,ch1,max:string;
    i,j:integer;
(*****)
Function palindrome(ch:string):boolean;
var i:integer;
    verif:boolean;
begin
  i:=0;
  repeat
    i:=i+1;
    verif := (ch[i] = ch[length(ch)-i+1]);
  until (verif=false) or (i=length(ch) div 2);
  palindrome:=verif;
end;
(***** p.p *****)
begin
  write('Donner ch = ');readln(ch);
  max:=ch[1];
  repeat
    for j:=length(ch) downto 3 do
      begin
        ch1:=copy(ch,1,j);
        if palindrome(ch1) and (length(ch1)>length(max))
          then max:=ch1;
        end;
      delete(ch,1,1);
    until length(ch)=2;
  writeln ('La plus longue chaine palindrome est : ', max)
end.

```

STRUCTURE GENERALE D'UN PROGRAMME PASCAL

PROGRAM Nom-du-programme ; {En-tête du programme}*
Uses ... ; {Utilisation des unités / bibliothèques}*
Const ... ; {Déclaration des constantes}*
Type ... ; {Déclaration des types}*
Var ... ; {Déclaration des variables}*
{===== Définition des procédures =====}*
Procedure Nom-de-la-procédure (paramètres formels) ;
{Déclarations locales : Const, Type, Var, Function, Procedure, ...}*
Begin
Instructions de la procédure ;
End ;
{===== Définition des fonctions =====}*
Function Nom-de-la-fonction (paramètres formels) ; Type résultat ;
{Déclarations locales : Const, Type, Var, Function, Procedure, ...}*
Begin
Instructions de la fonction ;
Nom-de-la-fonction := résultat ;
End ;
{===== P. P. =====}
BEGIN {Début du programme principal}
Instruction ;
..... ;
{Bloc principal du programme avec appel des procédures et des fonctions}
END. {Fin du programme}

* : facultatif