



I- INSTRUCTION D'ENTRÉE ET DE SORTIE :

Solution Exercice 1 :

Algorithme

Algorithme conversion_temps;

Variable

S,M,H,T,R : entier;

Début

Écrire("Entrer T ");

Lire(T);

$H \leftarrow T/3600$;

$R \leftarrow \text{Mod}(T, 3600)$;

$M \leftarrow R/60$;

$S \leftarrow \text{Mod}(R, 60)$;

Écrire("T=", H, "h", M, "min", S, "s");

Fin

Programm Fortran

```
Program Conversion_temps
```

```
integer S,M,H,T,R
```

```
write(*,*) "Entrer T"
```

```
read(*,*) T
```

```
H = T/3600
```

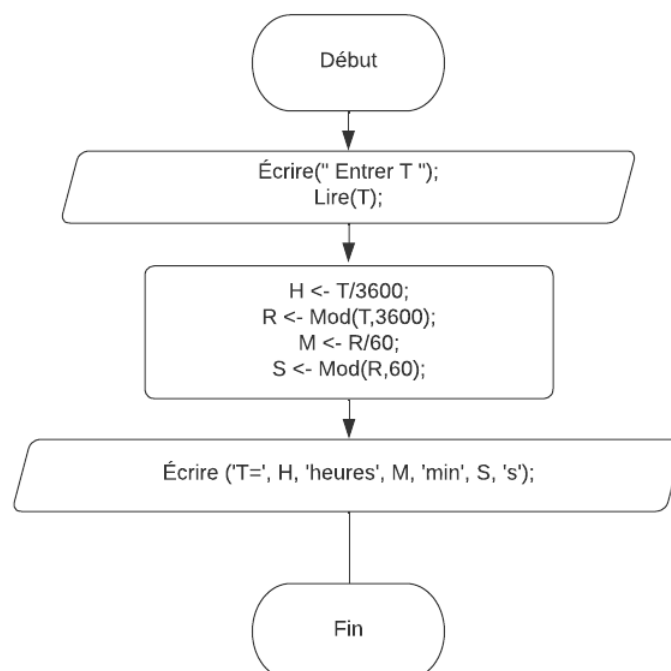
```
R = Mod(T,3600)
```

```
M = R/60
```

```
S = Mod(R,60)
```

```
write(*,*) "T=", H,"h", M,"min", S,"s"
```

```
end
```



Solution Exercice 2 :

Algorithme

Algorithme Conversion_dollar_dinar;

Variable

dollar, dinar : réel;

Début

Écrire("Entrer montant en Dinar ");

Lire(dinar);

$dollar \leftarrow dinar / 145,9;$

Écrire("Montant en Dollar =", dollar);

Fin

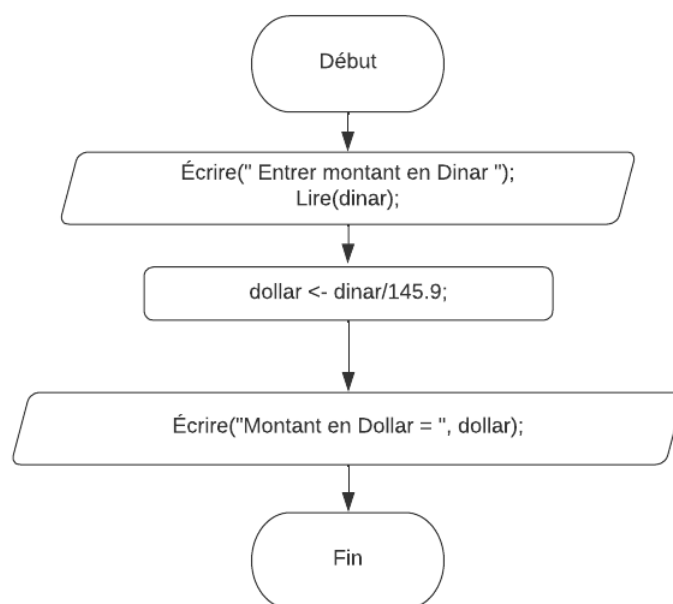
Programm Fortran

```
program Conversion_dollar_dinar
real dollar, dinar

write(*,*) "Entrer montant en Dinar"
read(*,*) dinar

dollar = dinar/145.9

write(*,*) "Montant en Dollar=", dollar
end
```



Solution Exercice 3 :

Algorithme

Algorithme Conversion_joule_calorie;

Variable

joule, cal : réel;

Début

Écrire("Entrer l'énergie en joule");

Lire(joule);

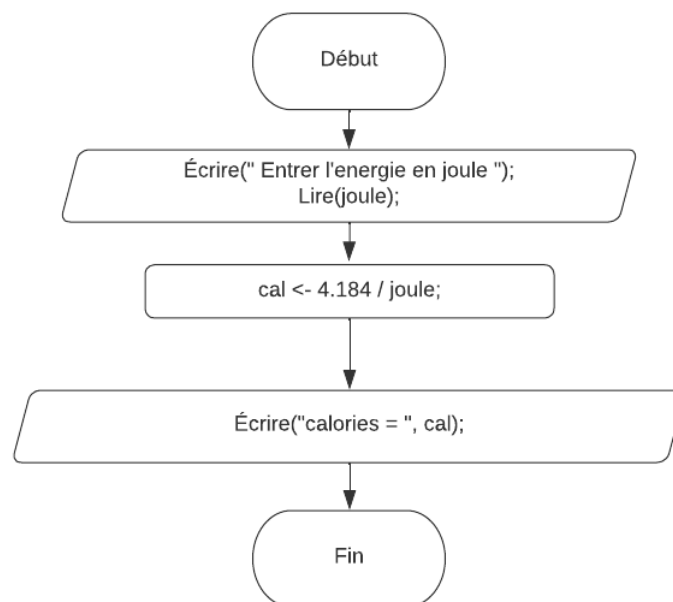
$cal \leftarrow 4,184/joule$;

Écrire("calories = ", cal);

Fin

Programm Fortran

```
program Conversion_joule_calorie  
  real joule, cal  
  write(*,*) "Entrer l'energie en joule"  
  read(*,*) joule  
  cal = 4.184/joule  
  write(*,*) "calories=", cal  
end
```



Solution Exercice 4 :

Algorithme

Algorithme Energie_Photon;

Variable

lamda, E : réel;

constante

$h = 4.13 \times 10^{-15}$, $c = 299792458$: réel;

Début

Écrire("Entrer l'énergie ");

Lire(E);

$\text{lamda} \leftarrow h * c / E$;

Écrire("lamda = ", lamda);

Fin

Programm Fortran

```
program Energie_Photon
  real :: h = 4.13*10e-15, c = 299792458
  real lamda, E

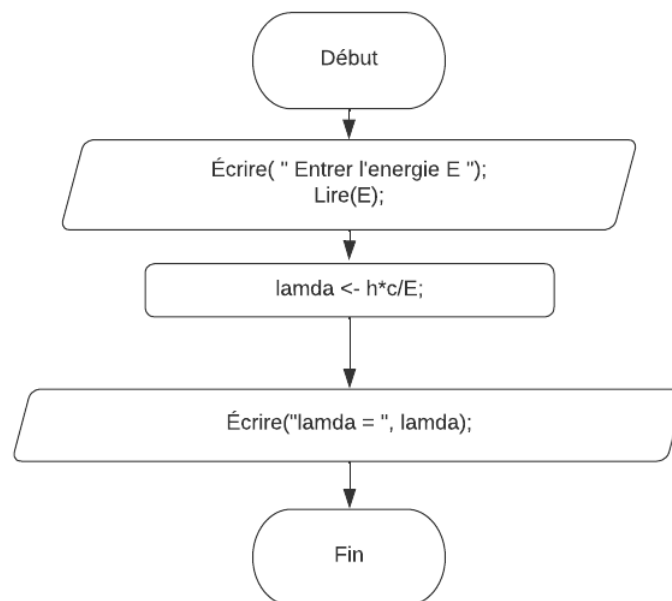
  write(*,*) "Entrer l'energie E"

  read(*,*) E

  lamda = h*c/E

  write(*,*) "lamda=", lamda

end
```



Solution Exercice 5 :

Algorithme

Algorithme Permutation;

Variable

a, b, c, x : entier;

Début

Écrire("Entrer a");

Lire(a);

Écrire("Entrer b");

Lire(b);

Écrire("Entrer c");

Lire(c);

$x \leftarrow a;$

$a \leftarrow b;$

$b \leftarrow c;$

$c \leftarrow x;$

Écrire("a = ", a);

Écrire("b = ", b);

Écrire("c = ", c);

Fin

Programm Fortran

```
program Permutation
integer a, b, c, x

write(*,*) "Entrer a"
read(*,*) a

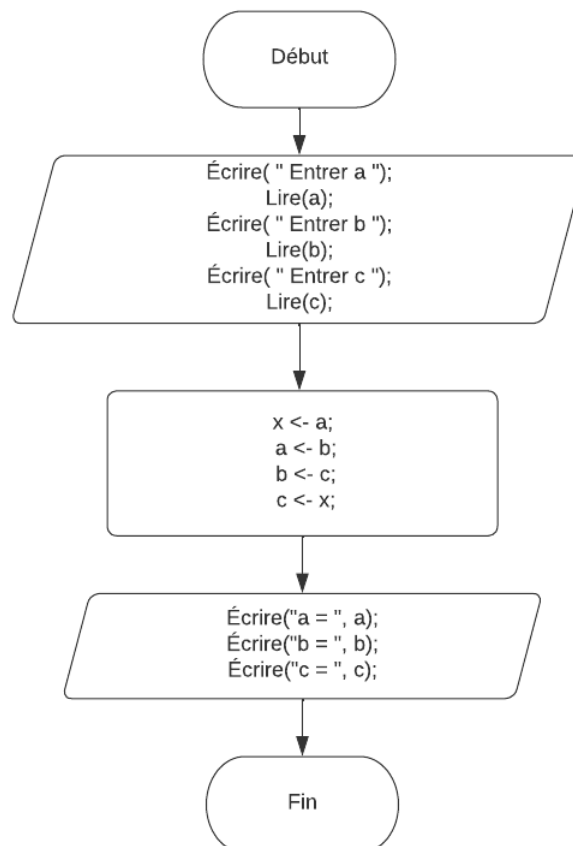
write(*,*) "Entrer b"
read(*,*) b

write(*,*) "Entrer c"
read(*,*) c

x=a
a=b
b=c
c=x

write(*,*) "a=", a
write(*,*) "b=", b
write(*,*) "c=", c

end
```



II- INSTRUCTION ALTERNATIVE (CONDITIONNELLE) :

Solution Exercice 1 :

Algorithme

Algorithme Minimum;

Variable

A, B, C, Min : Entier;

Début

Écrire("Entrer A");

Lire(A);

Écrire(" Entrer B");

Lire(B);

Écrire("Entrer C");

Lire(C);

$Min \leftarrow A$;

Si $B < Min$.and. $B < C$ Alors

$Min \leftarrow B$;

Écrire("minimum = ", Min);

Sinon

$Min \leftarrow C$;

Écrire("minimum = ", Min);

Fin Si

Fin

Programm Fortran

```
program minimum
integer A, B, C, Min

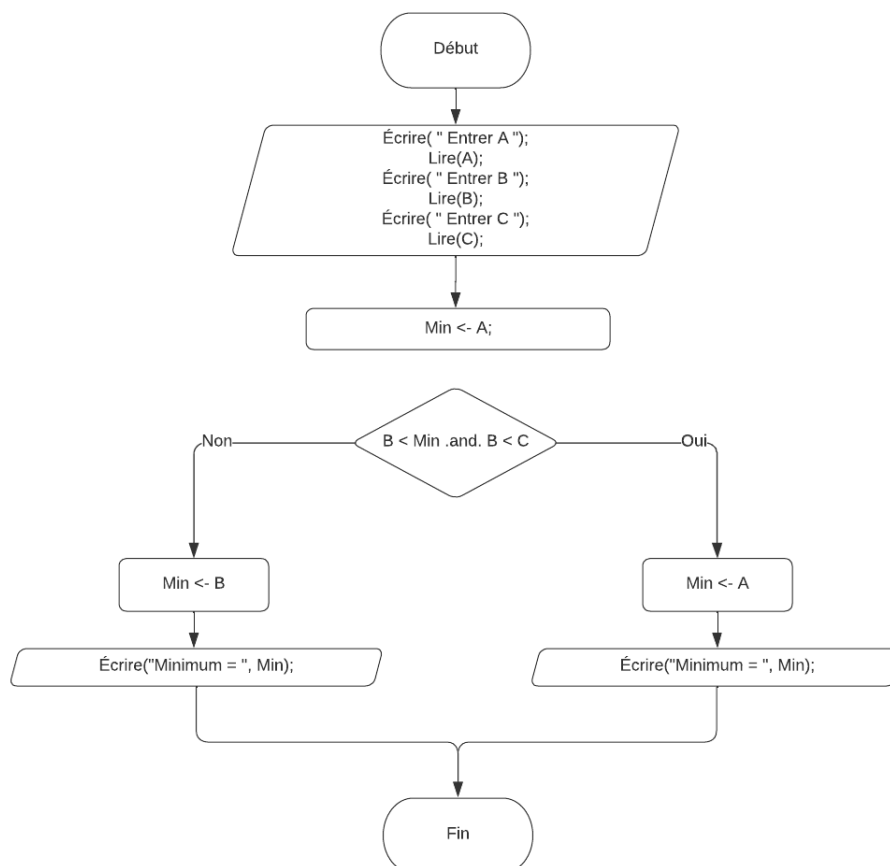
write(*,*) "Entrer A"
read(*,*) A

write(*,*) "Entrer B"
read(*,*) B

write(*,*) "Entrer C"
read(*,*) C

Min = A
if(B < Min .and. B < C)then
    Min = B
    write(*,*) "minimum=", Min
else
    Min = C
    write(*,*) "minimum=", Min
end if

end
```



Solution Exercice 2 :

Algorithme

Algorithme Egalite;

Variable

A, B : Entier;

Début

Écrire("Entrer A");

Lire(A);

Écrire(" Entrer B");

Lire(B);

Si A == B Alors

Écrire("A et B sont egaux");

Sinon

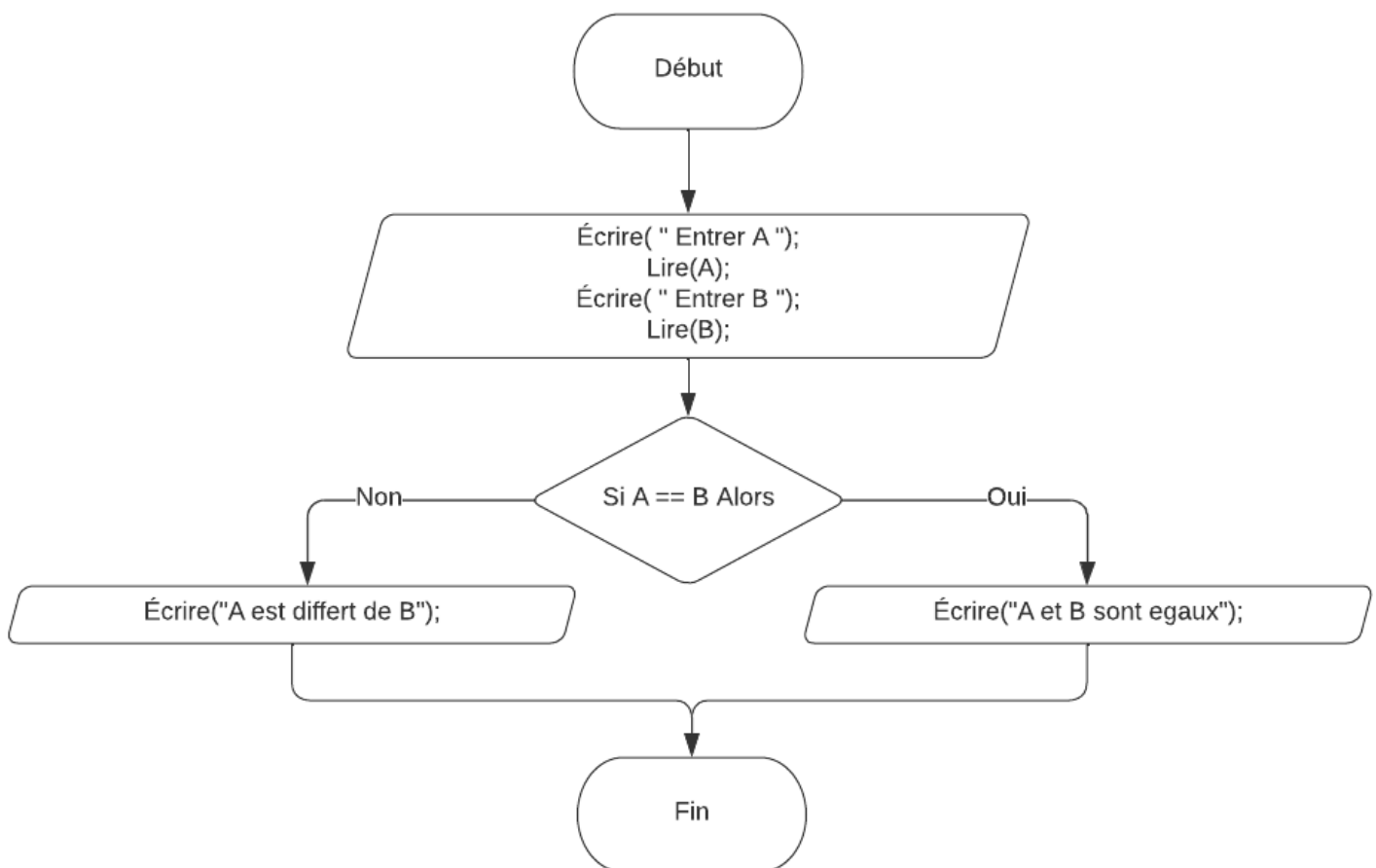
Écrire("A est differt de B");

Fin Si

Fin

Programm Fortran

```
program Egalite  
  
integer A, B  
  
write(*,*) "┐Entrer┐A"  
  
read(*,*) A  
  
write(*,*) "┐Entrer┐B"  
  
read(*,*) B  
  
if(A == B) then  
    write(*,*) "A┐et┐B┐sont┐egaux"  
else  
    write(*,*) "A┐est┐differt┐de┐B"  
end if  
  
end
```



Solution Exercice 3 :

Algorithme

Algorithme Parite;

Variable

X, P : Réel;

Début

Écrire("Entrer X");

Lire(X);

$P \leftarrow \text{Mod}(X, 2.);$

Si P .eq. 0 Alors

Écrire("X est pair");

Sinon

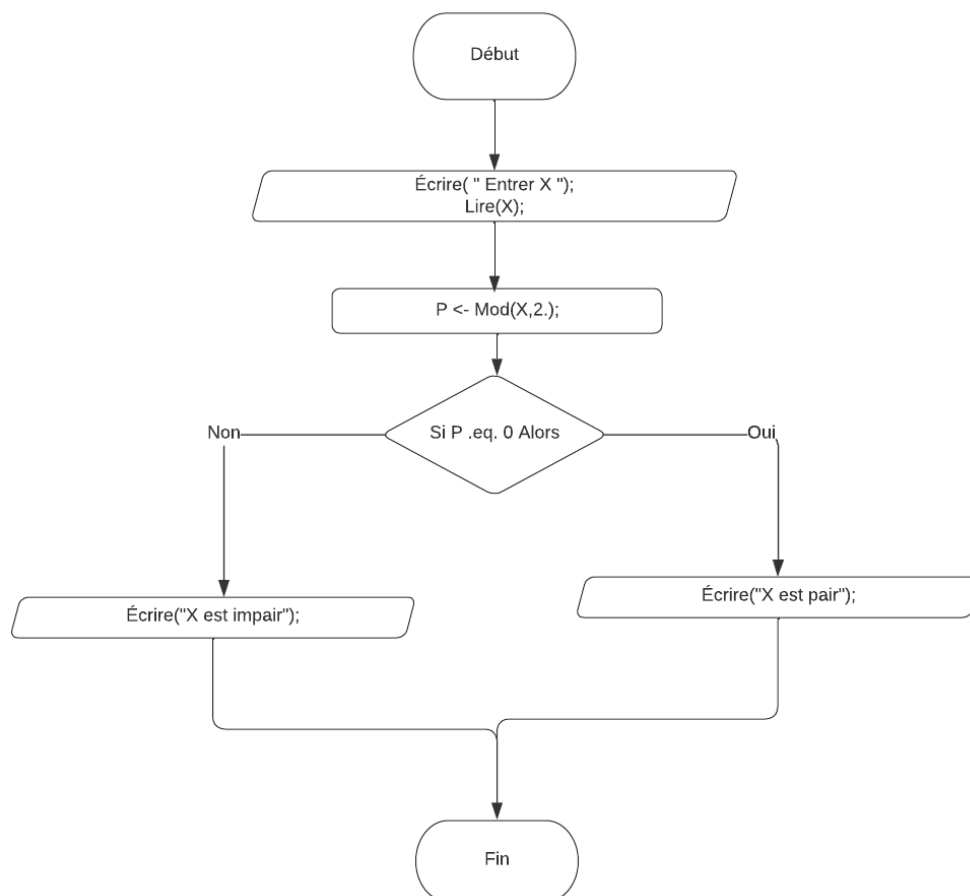
Écrire("X est impair");

Fin Si

Fin

Programm Fortran

```
program Parite
  real X, P
  write(*,*) "Entrer X"
  read(*,*) X
  P = Mod(X,2.)
  write(*,*) P
  if(P .eq. 0) then
    write(*,*) "X est pair"
  else
    write(*,*) "X est impair"
  end if
end
```



Solution Exercice 4 :

Algorithme

Algorithme Taille_Person;

Variable

Hauteur : Réel;

Taille : Chaîne de caractère;

Début

Écrire("Entrer hauteur en cm");

Lire(Hauteur);

Si Hauteur .lt. 150 Alors

$Taille \leftarrow "Nain";$

Sinon Si Hauteur .ge. 150 .and. Hauteur .lt. 165 Alors

$Taille \leftarrow "Taillemoyenne";$

Sinon

$Taille \leftarrow "Grand";$

Fin Si

Écrire("Taille est: ", Taille);

Fin

Solution Exercice 5 :

Algorithme

Algorithme E20;

Variable

delta, x0, x1, x2, a, b, c : Réel;

Début

Écrire("Entrer valeur de a,b,c");

Lire(a,b,c);

$delta \leftarrow b^2 - 4 * a * c;$

Écrire("delta=", delta);

Si delta < 0 Alors

Écrire("Solution Imaginaire");

Sinon Si delta == 0 Alors

$x0 \leftarrow -b/2 * a;$

Écrire("x0=", x0);

Sinon

$x1 \leftarrow -b - \sqrt{delta}/2 * a;$

$x2 \leftarrow -b + \sqrt{delta}/2 * a;$

Écrire("x1=", x1);

Écrire("x2=", x2);

Fin Si

Fin

Programm Fortran

```
program Taille_Person
```

```
real Hauteur
```

```
character*20 Taille
```

```
write(*,*) "Entrer_hauteur_en_cm"
```

```
read(*,*) Hauteur
```

```
if(Hauteur .lt. 150) then
```

```
    Taille = "Nain"
```

```
else if(Hauteur .ge. 150 .and.
```

```
        Hauteur .lt. 165) then
```

```
    Taille = "Taille_moyenne"
```

```
else
```

```
    Taille = "Grand"
```

```
end if
```

```
    write(*,*) "Taille_est:", Taille
```

```
end
```

Programm Fortran

```
program E20
```

```
real delta, x0, x1, x2, a, b, c
```

```
write(*,*) "Entrer_valeur_de_a,b,c"
```

```
read(*,*) a,b,c
```

```
delta = b**2 - 4*a*c
```

```
write(*,*) "delta=", delta
```

```
if(delta<0) then
```

```
    write(*,*) "solution_Imaginaire"
```

```
else if(delta == 0) then
```

```
    x0=-b/2*a
```

```
    write(*,*) "x0=", x0
```

```
else
```

```
    x1=-b-sqrt(delta)/2*a
```

```
    x2=-b+sqrt(delta)/2*a
```

```
    write(*,*) "x1=", x1
```

```
    write(*,*) "x2=", x2
```

```
end if
```

```
end
```

Solution Exercice 6 :

Algorithme

Algorithme Temperature;

Variable

T : Réel;

Début

Écrire("Entrer la temperature");

Lire(T);

Si $T < 0$ Alors

Écrire("Temps glacial");

Sinon Si $T == 0$.and. $T < 10$ Alors

Écrire("Temps tres froid");

Sinon Si $T \geq 0$.and. $T < 20$ Alors

Écrire("Temps froid");

Sinon Si $T \geq 20$.and. $T < 30$ Alors

Écrire("Temperature normale");

Sinon Si $T \geq 30$.and. $T < 40$ Alors

Écrire("Il fait chaud");

Sinon

Écrire("Il fait tres chaud");

Fin Si

Fin

Program Fortran

```
program Temperature
```

```
real T
```

```
write(*,*) "Entrer la temperature"
```

```
read(*,*) T
```

```
if(T < 0) then
```

```
    write(*,*) "Temps glacial"
```

```
else if(T == 0 .and. T < 10) then
```

```
    write(*,*) "Temps tres froid"
```

```
else if(T >= 0 .and. T < 20) then
```

```
    write(*,*) "Temps froid"
```

```
else if(T >= 20 .and. T < 30) then
```

```
    write(*,*) "Temperature normale"
```

```
else if(T >= 30 .and. T < 40) then
```

```
    write(*,*) "Il fait chaud"
```

```
else
```

```
    write(*,*) "Il fait tres chaud"
```

```
end if
```

```
end
```

III- INSTRUCTION ITÉRATIVE (RÉPÉTITIVE) :

Solution Exercice 1 : (BOUCLE POUR)

Algorithme

Algorithme Somme;

Variable

n, i, s : entier;

Début

Écrire("Entrer n ");

Lire(n);

$s \leftarrow 0$;

Pour i $\leftarrow$ allant 1 jusqu'à n avec pas 1 **Faire**

$s \leftarrow s + i$;

Écrire("s=", s);

Fin Pour

Fin

BOUCLE TANT QUE :

Algorithme

Algorithme Somme;

Variable

n, i, s : entier;

Début

Écrire("Entrer n ");

Lire(n);

$s \leftarrow 0$;

$i \leftarrow 1$;

Tant que i $\leq$ n **Faire**

$s \leftarrow s + i$;

$i \leftarrow i + 1$;

Fin Tant que

Écrire("s=", s);

Fin

Programm Fortran

```
program Somme
integer n, i, s
write(*,*) "Entrer_valeur_de_n"
read(*,*) n
s=0
do i = 1, n, 1
    s=s+i
write(*,*) "s=", s
end do
end
```

Programm Fortran

```
program Somme
integer n, i, s
write(*,*) "Entrer_valeur_de_n"
read(*,*) n
s=0
i=1
do while(i <= n)
    s=s+i
    i=i+1
end do
write(*,*) "s=", s
end
```

Solution Exercice 2 : (BOUCLE POUR)

Algorithme

Algorithme Puissance;

Variable

n, i : Entier;

x, p : Réel;

Début

Écrire("Entrer valeur de x et n ");

Lire(n);

$p \leftarrow 1$;

Pour i $\leftarrow$ allant 1 jusqu'à n avec pas 1 **Faire**

$p \leftarrow p * x$;

Écrire("p=", p);

Fin Pour

Fin

BOUCLE TANT QUE : Algorithme

Algorithme Puissance;

Variable

n, i, s : entier;

Début

Écrire("Entrer n ");

Lire(n);

$p \leftarrow 1$;

$i \leftarrow 1$;

Tant que i $\leq$ n **Faire**

$p \leftarrow p * x$;

$i \leftarrow i + 1$;

Écrire("p=", p);

Fin Tant que

Fin

Programm Fortran

```
program Puissance
```

```
real x,p
```

```
integer n, i
```

```
write(*,*) "Entrer_valeur_de_x_et_n"
```

```
read(*,*) x, n
```

```
p = 1
```

```
do i = 1, n, 1
```

```
    p=p*x
```

```
    write(*,*) "p=", p
```

```
end do
```

```
end
```

Programm Fortran

```
program Puissance
```

```
real x,p
```

```
integer n, i
```

```
write(*,*) "Entrer_valeur_de_x_et_n"
```

```
read(*,*) x, n
```

```
p = 1
```

```
i=1
```

```
do while(i <= n)
```

```
    p=p*x
```

```
    i=i+1
```

```
    write(*,*) "p=", p
```

```
end do
```

```
end
```

Solution Exercice 3 : (BOUCLE POUR)

Algorithme

Algorithme Factorielle;

Variable

n, i, fact : Entier;

Début

Écrire("Entrer valeur de n ");

Lire(n);

$fact \leftarrow 1$;

Pour i $\leftarrow$ allant 1 jusqu'à n avec pas 1 **Faire**

$fact \leftarrow fact * i$;

Écrire("fact=", fact);

Fin Pour

Fin

BOUCLE TANT QUE :

Algorithme

Algorithme Factorielle;

Variable

n, i, fact : Entier;

Début

Écrire("Entrer valeur de n ");

Lire(n);

$i \leftarrow 1$;

$fact \leftarrow 1$;

Tant que i $\leq$ n **Faire**

$fact \leftarrow fact * i$;

$i \leftarrow i + 1$;

Écrire("fact=", fact);

Fin Pour

Fin

Programm Fortran

```
program Factorielle
integer n, i, fact
write(*,*) "Entrer_valeur_de_n"
read(*,*) n
fact = 1
do i = 1, n, 1
    fact = fact * i
write(*,*) "fact=", fact
end do
end
```

Programm Fortran

```
program Factorielle
integer n, i, fact
write(*,*) "Entrer_valeur_de_n"
read(*,*) n
fact = 1
i=1
do while(i <= n)
    fact = fact*i
    i = i+1
    write(*,*) "fact=", fact
end do
end
```

Solution Exercice 5 : (BOUCLE POUR)

Algorithme

Algorithme Suite1;

Variable

n, i : Entier;

$frac$: Réel;

Début

Écrire("Entrer valeur de n ");

Lire(n);

$frac \leftarrow 0$;

Pour $i \leftarrow$ allant 1 jusqu'à n avec pas 1 **Faire**

$frac \leftarrow frac + 1./i$;

Écrire("frac=", $frac$);

Fin Pour

Fin

Algorithme

Algorithme Factorielle;

Variable

$n, i, fact$: Entier;

$frac$: Réel;

Début

Écrire("Entrer valeur de n ");

Lire(n);

$frac \leftarrow 0$;

$fact \leftarrow 1$;

Pour $i =$ allant de 1 jusqu'à n **Faire**

$fact \leftarrow fact * i$;

$frac \leftarrow frac + 1./fact$;

Écrire("frac=", $frac$);

Fin Pour

Fin

Programm Fortran

```
program Suite1
integer n, i
real frac

write(*,*) "Entrer_valeur_de_n"

read(*,*) n

frac = 0

do i = 1, n, 1

    frac = frac + 1./i

write(*,*) "frac=", frac

end do

end
```

Programm Fortran

```
program Suite2
integer n, i, fact
real frac

write(*,*) "Entrer_valeur_de_n"

read(*,*) n

frac = 0
fact = 1

do i = 1, n, 1

    fact = fact * i

    frac = frac + 1./fact

write(*,*) "frac=", frac

end do

end
```

Algorithme

Algorithme Suite3;

Variable

n, i : Entier;
frac : Réel;

Début

Écrire("Entrer valeur de n ");
Lire(n);

$frac \leftarrow 0$;
 $fact \leftarrow 1$;

Pour i $\leftarrow$ allant 1 jusqu'à n avec pas 1 **Faire**

$fact \leftarrow fact * i$;
 $frac \leftarrow frac + 1. * i / fact$;

Écrire("frac=", frac);

Fin Pour

Fin

Algorithme

Algorithme Suite4;

Variable

n, i : Entier;
frac, x : Réel;

Début

Écrire("Entrer valeur de n et x ");
Lire(n, x);

$frac \leftarrow 0$;
 $fact \leftarrow 1$;

Pour i $\leftarrow$ allant 1 jusqu'à n avec pas 1 **Faire**

$fact \leftarrow fact * i$;
 $frac \leftarrow frac + x * i / fact$;

Écrire("frac=", frac);

Fin Pour

Fin

Programm Fortran

```
program Suite3

integer n, i, fact
real frac

write(*,*) "Entrer_valeur_de_n"

read(*,*) n

frac = 0
fact = 1

do i = 1, n, 1

    fact = fact * i

    frac = frac + 1.*i/fact

write(*,*) "frac=", frac

end do

end
```

Programm Fortran

```
program Suite4

integer n, i, fact
real frac, x

write(*,*) "Entrer_valeur_de_n_et_x"

read(*,*) n, x

frac = 0
fact = 1

do i = 1, n, 1

    fact = fact * i

    frac = frac + x*i/fact

write(*,*) "frac=", frac

end do

end
```