

CORRECTION
EXERCICES ALGORITHMME 1

Mr KHATORY
 (GIM 1° A)

1

EXERCICES ALGORITHMME

**Ecrire un algorithme permettant de résoudre une équation du second degré.
 Afficher les solutions !**

$$ax^2 + bx + c = 0; \quad \text{solution: } x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Solution:

ALGORITHMME seconddegré

VAR a, b, c, delta : REEL

DEBUT

ECRIRE (" saisissez les valeurs a, b et c de l'équation $ax^2+bx+c=0$: ")

LIRE (a, b, c)

SI (a=0)

ALORS

ECRIRE (" équation du premier degré ")

SI (b≠0)

ALORS **ECRIRE** ("solution est ", -c/b)

SINON **ECRIRE** (" Pas de solution")

FINSI

SINON

 delta \Leftarrow $b^2-4*a*c$

SI (delta > 0)

ALORS

ECRIRE ("les solutions sont ", $\frac{-b - \text{racine}(\text{delta})}{2 \bullet a}$ et ", $\frac{-b + \text{racine}(\text{delta})}{2 \bullet a}$)

SINON

SI delta =0 **ALORS** **ECRIRE** ("Solution est", -b/(2a))

SINON **ECRIRE** ("pas de solutions réelles !!")

FINSI

FINSI

FINSI

FIN

Fonction
standard

2

EXERCICES ALGORITHME

Ecrire le même algorithme avec des selon-que :

ALGORITHME seconddegré

VAR a, b, c, delta: REEL

DEBUT

 Ecrire ("saisissez les valeurs de a, b et c de l'équation ax^2+bx+c ")

 Lire (a, b, c)

 Si (a=0)

ALORS

 Ecrire ("équation du premier degré ")

SI (b<>0)

ALORS Ecrire ("solution est ", -c/b)

SINON Ecrire (" Pas de solution")

FINSI

SINON

 delta \Leftarrow $b^2-4*a*c$

SELONQUE

 delta = 0 : Ecrire ("la solution unique est:", -b/(2a))

 delta > 0 : Ecrire (" les deux solutions sont ", $\frac{-b - \text{racine}(\text{delta})}{2*a}$, " et " , $\frac{-b + \text{racine}(\text{delta})}{2*a}$)

SINON Ecrire (" pas de solution réelle ")

FINSELON

FINSI

FIN

3

EXERCICES ALGORITHME

Ecrire un algorithme qui donne la durée de vol en heure minute connaissant l'heure de départ et l'heure d'arrivée.

On considère que le départ et l'arrivée ont lieu le même jour !

Données: h1,m1,h2 et m2

On suppose que $h2 > h1$!!

Cas possibles pour m1 et m2 \longrightarrow 2 cas ($m1 < m2$ ou $m1 > m2$)

4

EXERCICES ALGORITHME

Ecrire un algorithme qui donne la durée de vol en heure minute connaissant l'heure de départ et l'heure d'arrivée.

On considère que le départ et l'arrivé ont lieu le même jour

Solution:

```

ALGORITHME DuréeVol
VAR h1, h2, m1, m2: ENTIER
    hd, md : ENTIER
DEBUT
    ECRIRE (" entrer horaire de départ: h min")
    LIRE (h1, m1)
    ECRIRE (" entrer horaire d'arrivée: h min")
    LIRE (h2, m2)
    SI (m2 > m1 )
        ALORS
            hd ⇐ h2-h1
            md ⇐ m2-m1
            ECRIRE (" la durée de vol est : ", hd ,':', md)
        SINON
            hd ⇐ h2-h1-1
            md ⇐ m2+60-m1
            ECRIRE (" la durée de vol est : ", hd ,':', md)
    FINSI
FIN
  
```

5

EXERCICES ALGORITHME

Ecrire un algorithme qui donne la durée de vol en heure minute connaissant l'heure de départ et l'heure d'arrivée.

On considère que le départ et l'arrivé ont lieu le même jour

Solution n° 2:

```

ALGORITHME DureeVol1
VAR h1, h2, m1, m2: ENTIER
    hd, md : ENTIER
DEBUT :
    ECRIRE (" entrer horaire de départ: h min")
    LIRE (h1, m1)
    ECRIRE (" entrer horaire d'arrivée: h min")
    LIRE (h2, m2)
    md ⇐ [h2*60+m2] - [h1*60+m1]
    hd ⇐ md div 60      (* division entière (/)* )
    md ⇐ md mod 60    (*reste de la division entière (%)* )
    ECRIRE (" la durée de vol est : ", hd ,':', md)
FIN
  
```

6

EXERCICES ALGORITHME

On suppose que la durée de vol est inférieure à 24 heures mais peut avoir lieu le lendemain.

Etudier les différents cas ! 

Données: h_1, m_1, h_2 et m_2

Exemple1:
Départ :8h23 min
Arrivée: 13h 30 min

(* $m_1 < m_2$ *)

$h_1 < h_2$

Exemple2:
Départ :8h23 min
Arrivée: 13h 15 min

(* $m_1 > m_2$ *)

Exemple3:
Départ :17h30 min
Arrivée: 2h 40 min

(* $m_1 < m_2$ *)

$h_1 > h_2$

Exemple4:
Départ :17h30 min
Arrivée: 2 h 25 min

(* $m_1 > m_2$ *)

➤ Comparer h_1 et h_2 ! (2 cas)

➤ Pour chaque cas: comparer m_1 et m_2 ! (2 cas)

 4 cas en tout !!

7

EXERCICES ALGORITHME

On suppose que la durée de vol est inférieure à 24 heures mais peut avoir lieu le lendemain.

ALGORITHME DureeVol2

VAR h_1, h_2, m_1, m_2 : ENTIER

hd, md : ENTIER

DEBUT

ECRIRE (" entrer horaire de départ et d'arrivée: h_1 m_1 h_2 m_2 ")

LIRE (h_1, m_1, h_2, m_2)

SI ($h_2 > h_1$)

ALORS

SI ($m_2 > m_1$)

ALORS

hd \Leftarrow $h_2 - h_1$
md \Leftarrow $m_2 - m_1$
ECRIRE (hd, md)

SINON

hd \Leftarrow $h_2 - h_1 - 1$
md \Leftarrow $m_2 + 60 - m_1$
ECRIRE (hd, md)

FINSI

SINON

SI ($m_2 > m_1$)

ALORS

hd \Leftarrow $h_2 - h_1 + 24$
md \Leftarrow $m_2 - m_1$
ECRIRE (hd, md)

SINON

hd \Leftarrow $h_2 - h_1 + 24 - 1$
md \Leftarrow $m_2 + 60 - m_1$
ECRIRE (hd, md)

FINSI

FINSI

FIN

Exemple:
Départ :8h23 min
Arrivée: 13h 30 min

Exemple:
Départ :8h23min
Arrivée: 13h 15 min

Exemple:
Départ :17h30min
Arrivée: 2h 40min

Exemple:
Départ :17h30min
Arrivée: 2h 25 min

8

EXERCICES ALGORITHMME

Écrire un algorithme qui lit trois valeurs entières (A, B et C) et qui permet de les trier par échanges successifs Et enfin les afficher dans l'ordre

ALGORITHME TriSuccessif

VAR A, B, C : ENTIER

DEBUT

ECRIRE (" entrer Les valeurs A , B et C ")

LIRE(A,B,C)

SI (A > B) ALORS

échange (A,B)

SI B > C ALORS

échange (B,C)

SI A > B ALORS

échange (A,B)

FINSI

FINSI

SINON

SI B > C ALORS

échange (B,C)

SI A > B ALORS

échange (A,B)

FINSI

FINSI

FINSI

ECRIRE ("Les valeurs A , B et C sont (d

FIN

ici A < B reste à vérifier B ? C

ici B < C ET A < C (reste A ? B)

Finalemnt A < B < C

Ici B < C ET A < C ((reste A ? B)

Ici A < B ET B < C ⇒ A < B < C

Finalemnt A < B < C

9

EXERCICES ALGORITHMME

Écrire un algorithme calculatrice permettant la saisie du premier entier (a) de l'opération (+ ou - ou * ou / : sont des caractères) et du deuxième entier (b) et qui affiche le résultat

ALGORITHME calculatrice

VAR a, b : ENTIER

op : CARACTERE

DEBUT

ECRIRE (" saisissez le premier entier ")

LIRE (a)

ECRIRE (" saisissez l'opération ")

LIRE (op)

ECRIRE (" saisissez le deuxième entier")

LIRE (b)

SELONQUE :

Op = '+' : ECRIRE ("la somme de ",a, "et de ",b, "est égale",a+b)

Op = '*' : ECRIRE ("le produit de ",a, "et de ",b, "est égale",a*b)

Op = '/' : SI (b=0) ALORS ECRIRE (" division impossible ")

SINON ECRIRE ("la division de ",a, "par ",b, "est égale", a/b)

FINSI

Op = '-' : ECRIRE ("la soustraction de ",a, "et de ",b, "est égale", a-b)

SINON: ECRIRE(" Opération invalide ")

FINSELONQUE

FIN

10

EXERCICES ALGORITHME

1. Ecrire un algorithme qui demande un nombre de départ, et qui calcule la **somme** des entiers jusqu'à ce nombre. Par exemple si l'on tape 4, l'algorithme doit calculer: $1 + 2 + 3 + 4 = 10$

BOUCLE POUR

ALGORITHME Somme_Nombres
 VAR i, S : ENTIER
 val : ENTIER
DEBUT
 ECRIRE (" Entrer un nombre")
 LIRE (val)
 S \leftarrow 0
POUR i **DE** 1 **A** val **FAIRE**
 S \leftarrow S+i
FINPOUR
 ECRIRE (" La somme des nombres de
 1 à ", val, "est ", S)
FIN

Equivalent
POUR

BOUCLE TANT QUE

Algorithm Somme_Nombres
 Var i, S : ENTIER
 Val : ENTIER
DEBUT
 ECRIRE (" Entrer un nombre entier:")
 LIRE(val)
 S \leftarrow 0
 i \leftarrow 1
TANTQUE i \leq val
FAIRE
 S \leftarrow S+i
 i \leftarrow i+1
FINTANTQUE
 ECRIRE (" La somme des nombres de 1 à ",
 val, "est ", S)
FIN

11

EXERCICES ALGORITHME

1. Ecrire un algorithme qui demande un nombre de départ, et qui calcule la **moyenne** des entiers jusqu'à ce nombre. Par exemple si l'on tape 4, l'algorithme doit calculer: $1 + 2 + 3 + 4 = 10 / 4 = 2.5$

BOUCLE POUR

ALGORITHME Moyenne_Nombres
 Var i, S : ENTIER
 Val : ENTIER
 Moyenne : REEL
DEBUT
 S \leftarrow 0
 LIRE (val)
POUR i **DE** 1 **A** val **FAIRE**
 S \leftarrow S+i
FINPOUR
 Moyenne \leftarrow S / val
 ECRIRE (" La moyenne des nombres de 1 à
 ", val, "est ", Moyenne)
FIN

Equivalent
POUR

BOUCLE TANT QUE

ALGORITHME Moyenne_Nombres
 Var i, S : ENTIER
 Val : ENTIER
 Moyenne : REEL
DEBUT
 S \leftarrow 0
 i \leftarrow 1
 Lire(val)
TANTQUE i \leq val
FAIRE
 S \leftarrow S+i
 i \leftarrow i+1
FINTANTQUE
 Moyenne \leftarrow S / val
 Ecrire (" La moyenne des nombres de
 1 à ", val, "est ", Moyenne)
FIN

12

EXERCICES ALGORITHMME

Ecrire l'algorithme qui affiche la somme des prix d'une suite d'articles en DH (entiers) saisies par l'utilisateur et **se terminant par zéro**.

BOUCLE TANTQUE

```
ALGORITHME Somme_Prix
VAR p, S : ENTIER
DEBUT
S ← 0
ECRIRE("Entrer le prix du 1° article:")
LIRE(p)
TANTQUE (p≠0)
FAIRE
  S ← S+p
  ECRIRE("Entrer le prix de l'article suivant( 0 si Fin):")
  LIRE(p)
FINTANTQUE
ECRIRE (" La somme des prix des articles est ", S)
FIN
```

BOUCLE REPETER

```
ALGORITHME Somme_Prix
VAR p, S : ENTIER
DEBUT
S ← 0
ECRIRE("Entrer le prix du 1° article:")
LIRE(p)
REPETER
  S ← S+p
  ECRIRE("Entrer le prix de l'article suivant( 0 si
  Fin):")
  LIRE(p)
JUSQU'A (p =0)
ECRIRE (" La somme des prix des articles est ", S)
FIN
```

13

EXERCICES ALGORITHMME

Ecrire l'algorithme qui affiche la somme des prix d'une suite d'articles en DH (entiers) saisies par l'utilisateur et **se terminant par zéro**.

Cas d'entrée à la boucle:

Si au départ $p=0$

→ choisir la boucle **TANTQUE**

Cas particulier (p=0) (Boucle **REPETER**) :
 On peut changer l'algorithme :

```
ALGORITHME Somme_Prix
VAR p, S : ENTIER
DEBUT
S ← 0
REPETER
  ECRIRE("Entrer le prix de l'article ( 0 si Fin):")
  LIRE(p)
  S ← S+p
JUSQU'A p=0
ECRIRE (" La somme des prix des articles est ", S)
FIN
```



```
ALGORITHME Somme_Prix
VAR p, S : ENTIER
DEBUT
S ← 0
ECRIRE("Entrer le prix du 1° article:")
LIRE(p)
REPETER
  S ← S+p
  ECRIRE("Entrer le prix de l'article suivant( 0 si
  Fin):")
  LIRE(p)
JUSQU'A p=0
ECRIRE (" La somme des prix des articles est ", S)
FIN
```

14

EXERCICES ALGORITHME

Écrire un algorithme qui demande successivement 10 nombres à l'utilisateur, et qui affiche à la fin le plus grand de ces 10 nombres Et aussi son rang

Exemple :

Entrez le nombre numéro 1 : 13
 Entrez le nombre numéro 2 : 17

 Entrez le nombre numéro 10 : 5

Le plus grand de ces nombres est : 17
 C'était le 2 ème nombre saisi

15

EXERCICES ALGORITHME

ALGORITHME Somme_10Nombres

```

CONST  NBRE=10;
VAR    indice , val : ENTIER
        Indice_grand, PLUSGRAND :ENTIER

DEBUT
  ECRIRE("Entrez le 1er nombre : ")
  LIRE (Val)
  PLUSGRAND ⇐ val
  Indice_grand ⇐ 1
  indice ⇐ 2
  TANTQUE (indice ≤ NBRE)
  FAIRE
    ECRIRE("Entrez le nombre numéro : ", indice)
    LIRE (Val)
    SI (val > PLUSGRAND) ALORS
      indice_grand ⇐ indice
      PLUSGRAND ⇐ val
    FINSI
    indice ⇐ indice+1
  FINTANTQUE
  ECRIRE("le plus grand de ces nombres est.", PLUSGRAND)
  ECRIRE(" c'était le ",indice_grand, " ème nombre saisi)

```

POUR indice DE 2 A NBRE

```

  Ecrire("Entrez le nombre numéro", indice)
  Lire (Val)
  Si val > PLUSGRAND alors
    Indice_grand ⇐ indice
    PLUSGRAND ⇐ val

```

Finsi

FINPOUR

FIN

16

EXERCICES ALGORITHMME

Ecrire un programme mettant en œuvre le jeu suivant :

Le premier utilisateur saisi un entier que le second doit deviner. Pour cela, il a le droit à autant de tentatives qu'il souhaite. A chaque échec, le programme lui indique si l'entier cherché est plus grand ou plus petit que sa proposition.

Un score indiquant le nombre de coups joués est mis à jour et affiché lorsque l'entier est trouvé.

ALGORITHME devinette

VAR a, n, t : ENTIER

DEBUT

ECRIRE(" Entrez le nombre à deviner")

LIRE (a)

ECRIRE("entrez le nombre (premier essai")

LIRE (n)

t ← 1

TANTQUE (a ≠ n)

FAIRE

SI n > a **ALORS** ECRIRE (" nombre cherché plus petit que : ",n)

SINON ECRIRE (" nombre cherché plus grand que ",n)

FINSI

 t ← t+1

 ECRIRE("entrez un autre nombre (tentative N° ",t,"")

 LIRE (n)

FINTANTQUE

ECRIRE (" c'est gagné : le nombre de tentatives est" ,t)

FIN

17

EXERCICES ALGORITHMME

Ecrire un programme mettant en œuvre le jeu suivant :

Le premier utilisateur saisi un entier que le second doit deviner. Pour cela, il a le droit à autant de tentatives qu'il souhaite. A chaque échec, le programme lui indique si l'entier cherché est plus grand ou plus petit que sa proposition.

Un score indiquant le nombre de coups joués est mis à jour et affiché lorsque l'entier est trouvé.

ALGORITHME devinette

VAR a, n, t : ENTIER

DEBUT

ECRIRE(" Entrez le nombre à deviner")

LIRE (a)

t ← 1

REPETER

 ECRIRE("Entrez un nombre (tentative N° : ",t,"")

 LIRE (n)

SELONQUE

 n > a : ECRIRE (" nombre cherché plus petit que : ",n)

 n < a : ECRIRE (" nombre cherché plus grand que ",n)

 n = a : ECRIRE (" c'est gagné : le nombre de tentatives est" ,t)

FINSELONQUE

 t ← t+1

JUSQU'A (a=n)

FIN

BOUCLE REPETER

18

EXERCICES ALGORITHME

Écrire une fonction **F_PGCD** qui retourne le PGCD de deux nombres en utilisant l'astuce suivante: soustrait le plus petit des deux entiers du plus grand jusqu'à ce qu'ils soient égaux.

Exemple:

a=24

b=36

Le PGCD ??

Boucle :

1. $a < b$ ($24 < 36$) $\rightarrow b = 36 - 24 = 12$

2. $b < a$ ($12 < 24$) $\rightarrow a = 24 - 12 = 12$

$a = b = 12$ on s'arrête donc le PGCD est **12**

19

EXERCICES ALGORITHME

Écrire une fonction **F_PGCD** qui retourne le PGCD de deux nombres en utilisant l'astuce suivante: soustrait le plus petit des deux entiers du plus grand jusqu'à ce qu'ils soient égaux.

FONCTION F_PGCD(a,b :ENTIER) :ENTIER

DEBUT

REPETER

SELONQUE

$a > b : a \leftarrow a - b$

$a < b : b \leftarrow b - a$

FINSELON

JUSQU'À $a = b$

retourner(a)

FIN

20

EXERCICES ALGORITHME

Ecrire la même fonction en utilisant l'algorithme d'Euclide: Utiliser les structures TANTQUE puis REPETER JUSQU'A

Boucle TANT QUE :

```

FUNCTION F_PGCD(a,b :ENTIER) :ENTIER
VAR r :ENTIER
DEBUT
r ⇐ a%b /* reste de la division entière*/
TANTQUE r≠0
FAIRE
    a ⇐ b
    b ⇐ r
    r ⇐ a%b
FINTANTQUE
retourner(b)
FIN
  
```

Boucle REPETER JUSQU'A :

```

FUNCTION F_PGCD(a,b :ENTIER) :ENTIER
VAR r :ENTIER
DEBUT
REPETER
    r ⇐ a%b
    a ⇐ b
    b ⇐ r
JUSQU'A r=0
retourner(b)
FIN
  
```

21

Merci
&
FIN

22