

Examen de rattrapage – Informatique 2

Exercice 01 : [5 pts]

Soit V, un vecteur de N nombres entiers. Écrire un programme Pascal qui permet de vérifier si les composantes du vecteur V sont dans un ordre croissant ou non.

Exercice 02 : [8 pts]

Soit l'algorithme suivant :

Algorithme Exo2;

Variables

A : tableau [1..100,1..100] de réel;

i, j, N, X, Y : entier;

Z : réel ;

Début

{*-*-* Les entrées *-*-*}

Lire(N);

Lire (X,Y);

Pour i ← 1 à N **faire**

Pour j ← 1 à N **faire**

 Lire(A[i,j]);

Fin-Pour

Fin-Pour

{*-*-* Traitement *-*-*}

Pour i ← 1 à N **faire**

 Z ← A [X, i];

 A [X, i] ← A [Y, i];

 A [Y, i] ← Z;

Fin-Pour

{*-*-* Les sorties *-*-*}

Pour i ← 1 à N **faire**

Pour j ← 1 à N **faire**

 Write(A[i,j]);

Fin-Pour

Fin-Pour

Fin.

Questions :

1. Traduire l'algorithme en Programme PASCAL.
2. Dérouler le programme pour les valeurs suivantes : N = 3, X = 1, Y = 3 et

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

3. Déduire ce que fait le programme ?
4. Réécrire le programme en remplaçant la boucle *for* par la boucle *Repeat* dans la partie de traitement.

Exercice 03 : [7 pts]

Soit le programme PASCAL suivant :

Program Exo3;

var

t_celsius: integer;

t_fahrenheit: real;

..... Conversion (x: integer): real;

var

z: real;

Begin

z := (x*1.8) +32;

.....

End;

Begin

read(t_celsius);

t_fahrenheit := conversion(t_celsius);

writeln('La température est : ', t_celsius, '°C');

writeln('La température est : ', t_fahrenheit:10:2, '°F');

End.

Questions :

1. Identifier le type de sous-programme présent dans ce programme.
2. Réécrire le sous-programme en complétant avec ce qui manque.
3. Dérouler le programme pour t_celsius= 31.
4. Déduire ce que fait le programme ?
5. Réécrire le programme en transformant le sous-programme utilisé à l'autre type de sous-programme.

Bonne chance

Corrigé d'examen de rattrapage – Informatique 2

Solution de l'exercice 01 : [5 pts]

Soit V, un vecteur de N nombres entiers. Écrire un programme Pascal qui permet de vérifier si les composantes du vecteur V sont dans un ordre croissant ou non.

Program pascal	
Program exercice_1;	} 0.25 point
Var	
V: array [1..100] of integer;	} 0.5 point
N, i: integer;	
Test : boolean;	} 0.25 point
Begin	
Read (N);	
For i:= 1 to N do	} 0.5 point
Read (V[i]);	
Test := True;	} 0.25 point
i := 2;	} 0.25 point
While ((i <= n) and (test = True)) do	} 0.75 point : 0.25 point pour la syntaxe de While et 0.25 point pour la condition
Begin	
If (v[i] < v[i-1]) Then	} 1.75 points :
Test := false;	
i:=i+1;	
end;	
If (test = True) Then	
Writeln ('Le vecteur V est trié dans un ordre croissant !')	} 0.5 point :
Else	
Writeln ('Le vecteur V n'est pas trié dans un ordre croissant !');	} 0.25 point pour la syntaxe et 0.25 point pour l'affichage
End.	

Solution de l'exercice 02 : [8 pts]

1. Traduire l'algorithme en Programme PASCAL. (3 points)

Program pascal	
<pre> Program Exo2; Var A : array [1..100,1..100] of real; i, j, N, X, Y: integer; Z : real ; Begin {*-.* Les entrées *-.*} Read(N); Read (X,Y) ; For i:= 1 to N do For j:= 1 to N do Read(A[i,j]); {*-.* Traitement *-.*} For i:= 1 to N do Begin Z := A [X, i] ; A [X, i] := A [Y, i] ; A [Y, i] :=Z ; End ; {*-.* Les sorties *-.*} For i:= 1 to N do Begin Writeln() ; For j:= 1 to N do Write(A[i,j], ' '); End ; End. </pre>	<div style="display: flex; flex-direction: column; align-items: flex-end;"> <div style="margin-bottom: 10px;">} 0.25</div> <div style="margin-bottom: 10px;">} 0.25</div> <div style="margin-bottom: 10px;">} 0.25</div> <div style="margin-bottom: 10px;">} 0.5</div> <div style="margin-bottom: 10px;">} 0.25</div> <div style="margin-bottom: 10px;">} 0.75</div> <div style="margin-bottom: 10px;">} 0.25</div> </div>

2. Dérouler le programme pour les valeurs suivantes : N = 3, X= 1, Y = 3 et $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ (2.5 point)

Instructions	Variables							Affichage									
	N	i	j	A	x	y	z										
Read(N);	3	/	/	/	/	/	/										
Read (X,Y) ;	3	/	/	/	1	3	/										
For i:= 1 to N do For j:= 1 to N do Read(A[i,j]);	//	1	1	<table border="1" style="display: inline-table; border-collapse: collapse; text-align: left;"> <tr><td>1</td><td>2</td><td>3</td></tr> <tr><td>4</td><td>5</td><td>6</td></tr> <tr><td>7</td><td>8</td><td>9</td></tr> </table>	1	2	3	4	5	6	7	8	9	//	//	/	
1	2	3															
4	5	6															
7	8	9															
For i:= 1 begin Z := A [X, i] ; A [X, i] := A [Y, i] ; A [Y, i] :=Z ; End ;	//	1	/	<table border="1" style="display: inline-table; border-collapse: collapse; text-align: left;"> <tr><td>7</td><td>2</td><td>3</td></tr> <tr><td>4</td><td>5</td><td>6</td></tr> <tr><td>1</td><td>8</td><td>9</td></tr> </table>	7	2	3	4	5	6	1	8	9	//	//	7	
7	2	3															
4	5	6															
1	8	9															

For i:= 2 begin Z := A [X, i] ; A [X, i] := A [Y, i] ; A [Y, i] :=Z ; End ;	//	2	/	<table border="1"> <tr><td>7</td><td>8</td><td>3</td></tr> <tr><td>4</td><td>5</td><td>6</td></tr> <tr><td>1</td><td>2</td><td>9</td></tr> </table>	7	8	3	4	5	6	1	2	9	//	//	8		} 0.25									
7	8	3																									
4	5	6																									
1	2	9																									
For i:= 3 begin Z := A [X, i] ; A [X, i] := A [Y, i] ; A [Y, i] :=Z ; End ;	//	3	/	<table border="1"> <tr><td>7</td><td>8</td><td>9</td></tr> <tr><td>4</td><td>5</td><td>6</td></tr> <tr><td>1</td><td>2</td><td>3</td></tr> </table>	7	8	9	4	5	6	1	2	3	//	//	9		} 0.25									
7	8	9																									
4	5	6																									
1	2	3																									
For i:= 1 to N do Begin Writeln() ; For j:= 1 to N do Write(A[i,j], ' '); End ;	//			<table border="1"> <tr><td>7</td><td>8</td><td>9</td></tr> <tr><td>4</td><td>5</td><td>6</td></tr> <tr><td>1</td><td>2</td><td>3</td></tr> </table>	7	8	9	4	5	6	1	2	3	//	//	//	<table border="1"> <tr><td>7</td><td>8</td><td>9</td></tr> <tr><td>4</td><td>5</td><td>6</td></tr> <tr><td>1</td><td>2</td><td>3</td></tr> </table>	7	8	9	4	5	6	1	2	3	} 0.5
7	8	9																									
4	5	6																									
1	2	3																									
7	8	9																									
4	5	6																									
1	2	3																									

3. D duire ce que fait le programme ? (0.75 point)

Le programme permet de permuter entre la X^{ me} et la Y^{ me} lignes.

4. R crire le programme en rempla ant la boucle for par la boucle while dans la partie de traitement. (1.75 point)

Program pascal	
Program Exo2; Var A : array [1..100,1..100] of real; i, j, N, X, Y: integer; Z : real ; Begin {*-*-* Les entr�es *-*-*} Read(N); Read (X,Y) ; For i:= 1 to N do For j:= 1 to N do Read(A[i,j]); i:= 1 ; } 0.25 Repeat } 0.25 Z := A [X, i] ; A [X, i] := A [Y, i] ; A [Y, i] :=Z ; } 0.75 (0.25 pour chaque instruction) i := i+1 ; } 0.25 Until (i > N) ; } 0.25 {*-*-* Les sorties *-*-*} For i:= 1 to N do Begin Writeln() ; For j:= 1 to N do Write(A[i,j], ' '); End ; End.	

Solution de l'exercice 03 : [7 pts]

1. Identifier le type de sous-programme présent dans ce programme. (0.25 point)

Le sous-programme est une fonction.

2. Ré-écrire le sous-programme en complétant avec ce qui manque. (1 point)

Program pascal
<pre> Program Exo3; var t_celsius: integer; t_fahrenheit: real; Function Conversion (x: integer): real; }- 0.25 var z: real; Begin z: = (x*1.8) +32; Conversion := z; }- 0.75 End; Begin read(t_celsius); t_fahrenheit: = conversion(t_celsius); writeln('La température est : ', t_celsius, ' °C'); writeln('La température est : ', t_fahrenheit:10:2, ' °F'); End. </pre>

3. Dérouler le programme pour t_celsius= 31. (2.5 points)

Instructions	Prog. Principal		La fonction			Affichage
	t_celsius	t_fahrenheit	x	z	Conversion	
read(t_celsius);	31	/	/	/	/	(0.25 pt)
t_fahrenheit:=conversion(t_celsius); <i>(l'appel à la fonction conversion avec le paramètre t_celsius=31)</i>						
⇒ La transmission des parameters z: = (x*1.8) +32; Conversion := z;			31	87.8	87.80	(0.5 pt) (0.5 pt) (0.5 pt)
t_fahrenheit:=conversion(t_celsius);		87.80				(0.25 pt)
writeln('La température est : ', t_celsius, ' °C');					(0.25 pt) →	La température est: 31 °C
writeln('La température est : ', t_fahrenheit:10:2, ' °F');					(0.25 pt) →	La température est: 87.80 °F

4. Déduire ce que fait le programme ? (1 point)

Le programme fait appel à un sous-programme qui fait la conversion de la température de degré Celsius en fahrenheit.

5. Réécrire le programme en transformant le type de sous-programme utilisé à l'autre type de sous-programme. (2.25 points)

Program pascal

Program Exo3;

var

t_celsius: integer;

t_fahrenheit: real;

procedure Conversion (x: integer; var z: real);

var

Begin

z: = (x*1.8) +32;

End;

Begin

read(t_celsius);

conversion(t_celsius, t_fahrenheit); (1 point)

writeln('La température est : ', t_celsius, '°C');

writeln('La température est : ', t_fahrenheit:10:2, '°F');

End.

1.25 point :

0.25 : Procedure

0.25 : Var

0.25 : z : real

0.5 : suppression de l'instruction

Conversion :=z ;