

EXAMEN FINAL – INFORMATIQUE 2 – 2025-05-22 À 12H30

Questions de cours (4 pts)

- 1- Les vecteurs et les matrices sont de quel type ? C'est quoi la différence entre eux ?
- 2- C'est quoi la différence entre les procédures et les fonctions ?
- 3- C'est quoi le passage de paramètres par valeur et le passage de paramètres par variable ?

Exercice N°01 (10 pts)

Partie A - Vecteurs (6 pts) : Soit l'algorithme suivant :

Alogithme

Algorithme Exo1;
Variables
T : Tableau [1..100] de réel;
X : réel; i, i1, i2, N : entier;
Début
Lire(N, i1, i2);
Pour i ← 1 **à** N **faire**
Lire(T[i]);
Fin-Pour

X ← T[i1]; T[i1] ← T[i2]; T[i2] ← X;

Pour i ← 1 **à** N **faire**
Ecrire(T[i]);
Fin-Pour
Fin.

1- Traduire l'algorithme en programme PASCAL

2- Dérouler l'algorithme pour N=4, i1=2 et i2=4

T = [15 , 16 , 18.5 , -5.25]

3- Qu'est-ce qu'il fait l'algorithme ?

4- Modifier le programme PASCAL pour échanger les valeurs de chaque position impair avec la position paire qui le suit (1 avec 2, 3 avec 4, ...)

Partie B - Matrices (4 pts) : Soit A une matrice réelle de N lignes et M colonnes.

Écrire un seul algorithme dans lequel :

- On introduit les dimensions et les valeurs de A
- On introduit les valeurs de deux indices j1 et j2
- Permuter les valeurs de deux colonnes d'indices j1 et j2 de la matrice A
- Calculer la matrice B la transposée de A
- Afficher les résultats des traitements ci-dessus

Indication

$$A = \begin{bmatrix} 5 & 33 \\ 10 & 15 \\ -9 & -8 \end{bmatrix} \xrightarrow{\text{Transposée}} B = \begin{bmatrix} 5 & 10 & -9 \\ 33 & 15 & -8 \end{bmatrix}$$

Exercice N°02 (6 pts)

Soit le programme PASCAL suivant :

Programme PASCAL

Program Exo2;
Var
X, S : real; n : integer;

Function power(y:real ; m:integer) : real;
Var R : real ; i:integer;
Begin
R := 1;
For i := 1 **to** m **do**
R := R * y;
power := R;
End;

BEGIN {Début du programme principal}
Read(n, X);
S := Power(X, n-1) + Power(X, n+1);
Write(S:0:2);
END. {Fin du programme principal}

Questions :

- 1- Dans ce programme PASCAL
 - a- Quelles sont les variables globales ?
 - b- Quelles sont les paramètres formels de power ?
 - c- Quelles sont les paramètres effectifs de power ?
 - d- C'est quoi le rôle de la fonction power ?
- 2- Dérouler le programme pour n=2 et X=2
- 3- Réécrire le programme en transformant la fonction power à une procédure.

Bonne Chance

EXAMEN FINAL – INFORMATIQUE 2 – 2025-05-22 À 12H30

CORRIGÉ TYPE

Questions de cours (4 pts)

1- Les vecteurs et les matrices sont de quel type ? C'est quoi la différence entre eux ?

Les vecteurs et les matrices sont de type **TABLEAU (Array)** (0.5)

La différence entre les vecteurs et les matrices :

Vecteur

- Tableau à une dimension
- Un seul indice pour accéder à une case (0.5)
- Contient une seule rangée
- Une seule boucle pour réaliser un parcours

Matrice

- Tableau à deux dimensions
- Deux indices pour accéder à une case (0.5)
- Contient des lignes et des colonnes
- Deux boucles pour réaliser un parcours

Au moins 2 différences

2- C'est quoi la différence entre les procédures et les fonctions ?

Les procédures et les fonctions sont des sous-programmes

Procédure

- Déclarée avec le mot clé **procedure**
- N'a pas de type de données
- Son résultat est un paramètre de sortie (passage par valeur)
- L'appel est une instruction à part (0.75)

Fonction

- Déclarée avec le mot clé **function**
- Possède un type de données
- son corps (instruction) se termine par `nom_fonction := <résultat>`
- L'appel se fait dans une affectation (souvent) (0.75)

Au moins 3 différences

3- C'est quoi le passage de paramètres par valeur et le passage de paramètres par variable (ou adresse) ?

Dans la déclaration des sous-programmes (procédures et/ou fonctions), on peut déclarer un paramètre soit avec passage par valeur ou passage par variable (ou adresse).

Passage par valeur

- Paramètre d'entrée
- N'est pas précédé par aucun mot clé
- Dans l'appel, le paramètre effectif peut être une valeur fixe, constante, variable ou expression.

Passage par variable (adresse)

- Paramètre d'entrée/sortie (ou de sortie)
- Précédé par le mot clé **var**
- Dans l'appel, le paramètre effectif doit être obligatoirement une variable.

(0.5)

Au moins 2 différences

Exercice N°01 (10 pts)

Partie A - Vecteurs (6 pts) : Soit l'algorithme suivant :

Algorithme
Algorithme Exo1; <u>Variables</u> T : <u>Tableau</u> [1..100] <u>de réel</u> ; X : <u>réel</u> ; i, i1, i2, N : <u>entier</u> ; <u>Début</u> Lire(N, i1, i2); <u>Pour</u> i ← 1 <u>à</u> N <u>faire</u> Lire(T[i]); <u>Fin-Pour</u> X ← T[i1]; T[i1] ← T[i2]; T[i2] ← X; <u>Pour</u> i ← 1 <u>à</u> N <u>faire</u> Écrire(T[i]); <u>Fin-Pour</u> <u>Fin.</u>

- 1- Traduire l'algorithme en programme PASCAL (1.75)
- 2- Dérouler l'algorithme pour N=4, i1=2 et i2=4 (2)
T=[15 , 16 , 18.5 , -5.25]
- 3- Qu'est-ce qu'il fait l'algorithme ? (1)
- 4- Modifier le programme PASCAL pour échanger les valeurs de chaque position impair avec la position paire qui la suit (indice 1 avec indice 2, 3 avec 4, ...) (1.25)

1- Traduire l'algorithme en programme PASCAL (1.75)

Programme PASCAL
Program Exo1; <u>Var</u> T : Array[1..100] of real; X : real; i, i1, i2, N : integer; <u>Begin</u> Read(N, i1, i2); <u>For</u> i:= 1 <u>to</u> N <u>do</u> Read(T[i]); X := T[i1]; T[i1] := T[i2]; T[i2] := X; <u>for</u> i:= 1 <u>to</u> N <u>do</u> Write(T[i]); <u>End.</u>

Cliquez sur lien suivant pour exécuter en ligne le programme : <https://onlinegdb.com/SnjrLSXvG>

2- Déroulement (N=4, i1=2, i2=4 et T=[15 , 16 , 18.5 , -5.25]) (2)

Instructions	Variables						Affichage
	N	i1	i2	i	X	T	
- Entrée - Lire(N, i1, i2) Pour i ← 1 à N faire Lire(T[i]);		0.25					
	4	2	4	/	/	/	
	"	"	"	1...4	/	1 2 3 4 [15 , 16 , 18.5 , -5.25]	0.25
- Traitement - X ← T[i1] = T[2] = 16 T[i1]←T[i2] : T[2]←T[4]=-5.25 T[i2]←X : T[4]←16					0.25 16	1 2 3 4 [15 , 16 , 18.5 , -5.25]	
	"	"	"			1 i1=2 3 i2=4 [15 , -5.25 , 18.5 , -5.25]	0.5
						1 i1=2 3 i2=4 [15 , -5.25 , 18.5 , 16]	0.5
- Sorties - Pour i ← 1 à N faire Écrire(T[i]);	"	"	"	1..4	"	"	0.25 1 2 3 4 [15 , -5.25 , 18.5 , 16]

3- Qu'est-ce qu'il fait l'algorithme ? (1)

L'algorithme permet de permuter (échanger) les valeurs de deux cases du vecteur T, les deux cases ont les indices i1 et i2 respectivement.

La réponse doit contenir l'une des expression :

- permuter deux cases
- échanger deux cases
- permutation entre deux cases
- (ou des expressions équivalentes)

4- Modifier l'algorithme pour échanger les valeurs de chaque position impair avec la position paire qui la suit (indice 1 avec indice 2, 3 avec 4, ...)

Programme PASCAL (variante 1)

```

Program Exo1_4;
Var
  T : Array[1..100] of real;
  X : real; i, N : integer;
Begin
  Read(N);
  For i:= 1 to N do
    Read( T[i] );

    //Permuter entre les positions (2i-1) et 2i
    for i:=1 to (N div 2) do
      if (2*i <= N) then
        begin
          X := T[2*i - 1];
          T[2*i - 1] := T[2*i];
          T[2*i] := X;
        end;

    for i := 1 to N do
      Write( T[i] );
End.

```

Programme PASCAL (variante 2)

```

Program Exo1_4;
Var
  T : Array[1..100] of real;
  X : real; i, N : integer;
Begin
  Read(N);
  For i:= 1 to N do
    Read( T[i] );

    //Permuter entre les positions i et (i+1)
    i := 1;
    while (i < N) do
      if (i < N) then
        begin
          X := T[i];
          T[i] := T[i+1];
          T[i+1] := X;
          i := i+2; //incrémenter i de 2
        end;

    for i := 1 to N do
      Write( T[i] );
End.

```

Soit correct soit faux
(1 ou 0)

Cliquez sur lien suivant pour exécuter en ligne le programme (variante O1) :

<https://onlinegdb.com/7O6Aq3SYi>

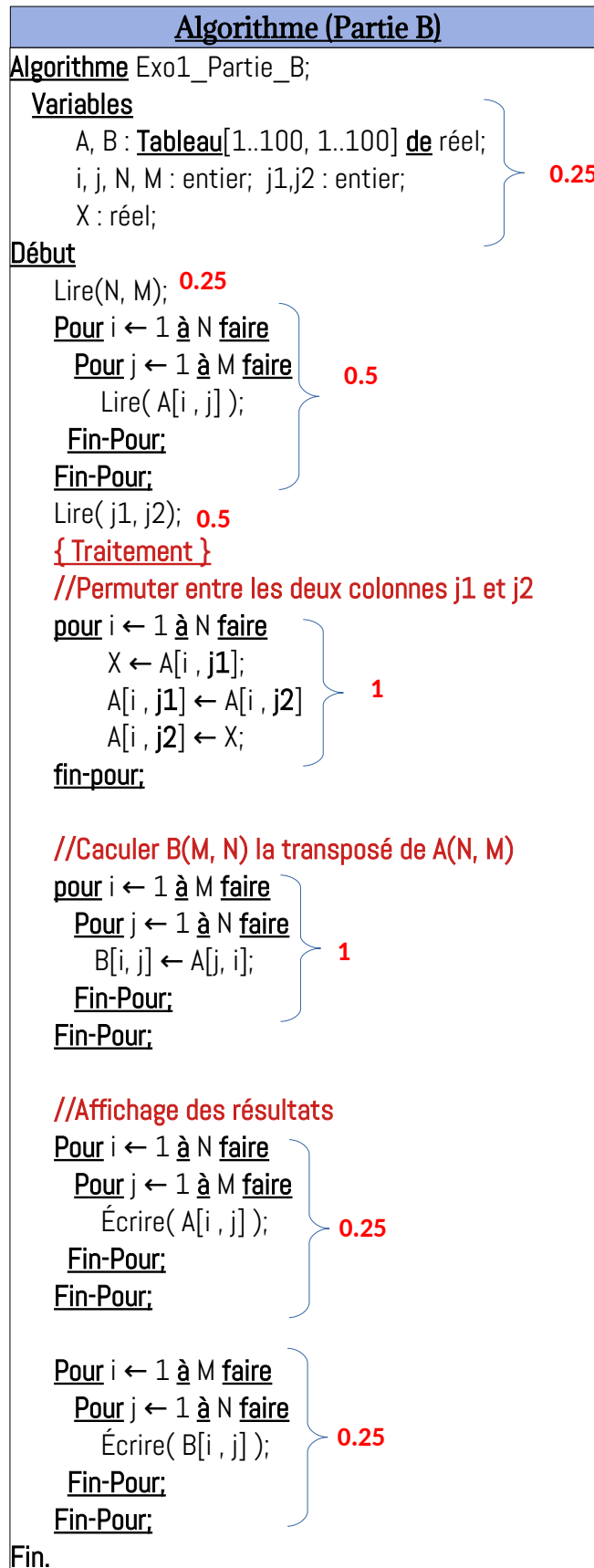
Cliquez sur lien suivant pour exécuter en ligne le programme (variante O2) :

<https://onlinegdb.com/EfzfNWFf2>

Partie B - Matrices (4 pts) : Soit A une matrice réelle de N lignes et M colonnes.

Écrire un seul algorithme dans lequel :

- On introduit les dimensions et les valeurs de la matrice A
- On introduit les valeurs de deux indices j1 et j2
- Permuter les valeurs des deux colonnes j1 et j2 de la matrice A
- Calculer la matrice B la transposée de A
- Afficher les résultats des traitements ci-dessus



Exercice N°02 (6 pts)

Soit le programme PASCAL suivant :

```

Programme PASCAL
Program Exo2;
  Var
    X, S : real;  n : integer;

  Function power(y:real ; m:integer) : real;
    Var R : real ; i:integer;
  Begin
    R:= 1;
    For i := 1 to m do
      R := R * y;
    power := R;
  End;

BEGIN {Début du programme principal}
  Read(n, X);
  S := Power(X, n-1) + Power(X, n+1);
  Write( S:0:2 );
END. {Fin du programme principal}
  
```

Questions :

- 1- Dans ce programme PASCAL 2
 - a- Quelles sont les variables globales ?
 - b- Quelles sont les paramètres formels de power ?
 - c- Quelles sont les paramètres effectifs de power ?
 - d- C'est quoi le rôle de la fonction power ?

- 2- Dérouler le programme pour $n=2$ et $X=2$ 2.5

- 1.5 3- Réécrire le programme en transformant la fonction power à une procédure.

- 1- Dans ce programme PASCAL 2

- 0.5a- Les variables globales sont : **X, S et n**
- 0.5b- Les paramètres formel de la fonction power sont : y (réel) et m (entier)
- c- Les paramètre effectifs de la fonction power sont :
- 0.25- Dans le premier appel Power(X, n-1) : les paramètre effectifs sont X et n-1
- 0.25- Dans le deuxième appel Power(X, n+1) : les paramètre effectifs sont X et n+1
- 0.5d- Le rôle de la fonction power est de calculer la valeur de Y^m (La puissance)

- 2- Dérouler le programme pour $n=2$ et $X=2$ 2.5

Instructions	Variables								Affichage
	Programme Principal			Fonction Power					
	X	n	S	y	m	i	R	Power	
Read(n, X);	2	0.25 2	/	Fonction Power n'est pas activée (état inactif du sous-programme)					
S:=power(X, n-1) + power(X, n+1); Le premier appel : power(X, n-1) X = 2 et n-1=2-1=1 ⇒ Powe(2, 1)									
- Appel Power(2, 1): - Transmission de paramètres y = 2 et m = 1 - Exécution des instructions R := 1; For i:= 1 R := y*i = 2*1 = 2 power := R = 2;				/	/	/	/	/	
				2	1				
						1	2	2 0.25	
⇒ Donc Power(2, 1) = 2									

Le deuxième appel : power(X, n+1) X = 2 et n+1=2+1=3 ⇒ Powe(2, 3)				état inactif du sous-programme					
- Appel Power(2, 3): - Transmission de paramètres y = 2 et m = 3 - Exécution des instructions R := 1; For i:= 1 R := y*i = 2*1 = 2 For i:= 2 R := y*i = 2*2 = 4 For i:= 3 R := y*i = 4*2 = 8 power := R = 8;				2	3				
						1	2		
						2	4		
						3	8	0.25	
								8	
⇒ Donc Power(2, 3) = 8				état inactif du sous-programme					
On remplace chaque appel par son résultat : S := power(X, n-1) + power(X, n+1) S := 2 + 8 S := 10									
			10						
Write(S:0:2)									10.00 0.25

3- Réécrire le programme en transformant la fonction Power à une procédure (1.5)

```

Programme PASCAL
Program Exo2_3;
Var
    X, S : real; n : integer; y1, y2:real;

    0.25
    Procedure power(y:real ; m:integer ; var R : real);
    Var i:integer;
    Begin
        R:= 1;
        For i := 1 to m do
            R := R * y;
            0.25 ← Supprimer
                    Power := R ;
    End;

BEGIN {Début du programme principal}
    Read(n, X);
    Power(X, n-1, y1); 0.25
    Power(X, n+1, y2); 0.25
    S := y1 + y2;

    Write( S:0:2 );
END. {Fin du programme principal}

```