République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de Bourse de formation Post-Graduée à l'Etranger
au titre de l'année universitaire 2009/2010
EPREUVE DE SPECIALITE : Mathématiques et Informatique(MI)

Matière : Algèbre 2 - Corrigé type sujet n°1
Durée : 45 mn

Exercice 1

1)

- $\mathbf{dim}_{IR}F \leq 4$. **(0,25pt)**
- $\mathbf{dim}_{IR}F_1 \leq 3$. **(0,25pt)**
- $\mathbf{dim}_{IR}F_2 \leq 3$. **(0,25pt)**

2) $\dim_{IR}F = 4 \Rightarrow \{v_1, v_2, v_3, v_4\}$ est libre. **(0,25pt)** d'où :
v_4 ne peut s'exprimer comme combinaison linéaire des vecteurs v_1, v_2 et v_3. **(0,5pt)**
v_1 ne peut s'exprimer comme combinaison linéaire des vecteurs v_2, v_3 et v_4. **(0,5pt)**
Alors $v_4 \notin F_1$ et $v_1 \notin F_2$.
Comme toute famille extraite d'une famille libre est libre **(0,5pt)** alors :
$\mathbf{dim}_{IR}F_1 = \mathbf{dim}_{IR}F_2 = 3$. **(0,5pt + 0,5pt)**

Exercice 2

1) Fausse **(0,25pt)** : car $0_E \in F \Rightarrow 0_E \notin C_E F$. **(0,25pt)**

2) Vraie **(0,25pt)** : car l'intersection contient au moins le vecteur nul 0_E. **(0,25pt)**

3) Fausse **(0,25pt)** : pour $F = \langle (0,1,2) \rangle$ et $G = \langle (0,1,2), (1,3,1) \rangle$,
$\dim F + \dim G = 1 + 2 = 3$.
Or $(0,1,2) \in F \cap G$ d'où $F \cap G \neq \{0\}$ par conséquent $F + G$, n'est pas une somme directe, il en résulte que F et G ne sont pas supplémentaires dans IR^3. **(0,25pt)**

4) Vraie **(0,25pt)** : car $\{1, i\}$ est une base de $\mathbb{C}$ sur IR. **(0,25pt)**

5) Vraie **(0,25pt)** : car $(0,6,1) = (1,2,0) + (-1,4,1)$ d'où
$\langle (1,2,0), (-1,4,1), (0,6,1) \rangle$ est contenu dans $\langle (1,2,0), (-1,4,1) \rangle$ d'où l'égalité. **(0,25pt)**

6) Fausse **(0,25pt)** : il est possible que A et B soient deux matrices non nulles, telles que $AB = 0$:
Pour $A = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$, $B = \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix}$ on a $A \neq 0$, $B \neq 0$ mais $AB = 0$. **(0,25pt)**

7) Vraie **(0,25pt)** : $\det(AA) = \det A \cdot \det A = (\det A)^2$, d'où le résultat par récurrence. **(0,25pt)**

8) Fausse **(0,25pt)** : un sous-espace vectoriel contient toujours le vecteur nul 0_E
Donc $\varnothing$ n'est pas un sous-espace vectoriel de E. **(0,25pt)**

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de Bourse de formation Post-Graduée à l'Etranger au titre de l'année universitaire 2009/2010

EPREUVE DE SPECIALITE: Mathématiques et Informatique (MI)
Matière : Algorithmique – Corrigé type sujet n° 1

<u>Exercice 1 (3.75 Pts)</u>

<u>Le principe de la méthode :</u>

Pour convertir un nombre (nb_base_x) écrit en base x :

1- Convertir nb_base_x en base 10, on obtient nb_base_10 ;
2- Si la base cible est égale à 10, le nombre est nb_base_10 ;
3- Sinon, convertir nb_base_10 en base y.

<u>1) Le programme en C</u>

```
#include<stdio.h>
main()
{int x,nb_base_x,y,puissance,nb_base_10,nb_base_y,n0,coefficient;
    printf("veuillez introduire une base et un nombre dans cette base\n");
    scanf("%d %d", &x,&nb_base_x);
    printf("veuillez introduire la base cible\n");
    scanf("%d",&y);
    /*conversion de base x en base 10*/
    n0=nb_base_x;
    nb_base_10=n0%10;
    puissance=1;
    while(n0!=0)
    {
      puissance=puissance*x;
      n0=n0/10;
      nb_base_10=nb_base_10+(n0%10)*puissance;
    }
   if (y==10)
    {
          nb_base_y=nb_base_10;
    }
    else
    {
      nb_base_y=0;
      coefficient=1;
      while(nb_base_10!=0)
      {
        nb_base_y=nb_base_y+(nb_base_10%y)*coefficient;
        coefficient=coefficient*10;
        nb_base_10=nb_base_10/y;
      }
```

```
   }
   printf("%d en base %d est égal à %d en base %d",nb_base_x,x,nb_base_y,y);
 }
```

2) Le programme en pascal

```
program exo1;
 var x,nb_base_x,y,puissance,nb_base_10,nb_base_y,n0,coefficient:integer;
 begin
    writeln('veuillez introduire une base et un nombre dans cette base');
    read(x,nb_base_x);
    writeln('veuillez introduire la base cible');
    read(y);
    {conversion de base x en base 10}
    n0:=nb_base_x;
    nb_base_10:=n0 mod 10;
    puissance:=1;
    while(n0<>0)do
    begin
      puissance:=puissance*x;
      n0:=n0 div 10;
      nb_base_10:=nb_base_10+(n0 mod 10)*puissance;

    end;

    if (y=10) then
    begin
          nb_base_y:=nb_base_10;
    end
    else
    begin
       nb_base_y:=0;
       coefficient:=1;
       while(nb_base_10<>0) do
       begin
         nb_base_y:=nb_base_y+(nb_base_10 mod y)*coefficient;
         coefficient:=coefficient*10;
         nb_base_10:=nb_base_10 div y;
       end;
    end;
    writeln(nb_base_x,'en base',x, 'est egal ...',nb_base_y, 'en base',y);
 end.
```

Exercice 2 (3.75 Pts)

Le principe de la méthode :

1. Parcourir le tableau à partir du deuxième élément ;
2. Déplacer en bulle chaque valeur supérieure à la valeur du premier élément ;
3. Mettre à jour l'indice du premier élément.

1) Le programme en C

```
#include<stdio.h>
main()
{int tab[5],i,j,temp,indice_premier;
for(i=0;i<5;i++)
{printf("donner une valeur\n");
scanf("%d",&tab[i]);
}
for(i=0;i<5;i++)
{
printf("%d    ",tab[i]);
}

/*Traitement*/
indice_premier=0;
i=1;
while(i<5)
{
 if (tab[i]>tab[indice_premier])
 { /*permutation*/
  j=i;
  while(j!=indice_premier)
  {
    temp=tab[j];
    tab[j]=tab[j-1];
    tab[j-1]=temp;
    j=j-1;
  }
  indice_premier=j+1;

 }
 i=i+1;
}

 for(i=0;i<5;i++)

printf("%d    ",tab[i]);
```

```
printf("Le nouvel indice de l'ancien premier élément est %d     ",indice_premier);
}
```

2) Le programme en pascal

```
program exo;
var tab:array[1..5] of integer;
   i,j,temp,indice_premier:integer;
begin
for i:=1 to 5 do
begin
writeln('donner une valeur');
readln(tab[i]);
end;

{Traitement}
indice_premier:=1;
i:=2;

while (i<=5) do
begin
 if (tab[i]>tab[indice_premier]) then
  begin
   j:=i;
   while(j<>indice_premier) do
   begin
     temp:=tab[j];
     tab[j]:=tab[j-1];
     tab[j-1]:=temp;
     j:=j-1;
   end;
   indice_premier:=j+1;

  end;
  i:=i+1;
end;

 for i:=1 to 5 do
begin
writeln(tab[i] , '      ');
end;
writeln('      le nouvel indice de l'ancien premier élément est ', indice_premier);

end.
```

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Concours National pour l'obtention de Bourse de formation Post-Graduée à l'Etranger au titre de l'année universitaire 2009/2010

EPREUVE DE SPECIALITE : Mathématiques et Informatique(MI)

Matière : Algèbre 2 - Corrigé type sujet n°2
Durée : 45 mn

Exercice 1

1) Soit $y \in \operatorname{Im} u^2$, il existe $x \in E$ tel que $u^2(x) = y$,
On a ainsi $y = u^2(x) = u(u(x))$,
Ce qui prouve que $y \in \operatorname{Im} u$. Ainsi : $\operatorname{Im} u^2 \subset \operatorname{Im} u$. **(1pt)**

2) Soit $x \in Ker\ u$, alors $u(x) = 0$, de plus : $u^2(x) = u(u(x)) = u(0) = 0$,
Ce qui prouve que $x \in Ker\ u^2$. Ainsi : $Ker\ u \subset Ker\ u^2$. **(1pt)**

Exercice 2

1) $A^2 = \begin{pmatrix} 1 & 0 \\ 3 & 4 \end{pmatrix} \quad AB = \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix} \quad B^2 = \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix}$

D'où $A^2 + 2AB + B^2 = \begin{pmatrix} 1 & 1 \\ 4 & 7 \end{pmatrix}$ **(0,5pt)**

Par ailleurs $(A+B)^2 = \begin{pmatrix} 0 & 1 \\ 2 & 2 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 2 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 2 \\ 4 & 6 \end{pmatrix}$ **(0,5pt)**

$(A+B)^2$ est différente de $A^2 + 2AB + B^2$.

2) Commentaire :
$(A+B)^2 \neq A^2 + 2AB + B^2$ est dû au fait que le produit matriciel n'est pas commutatif.
Comme $AB \neq BA$, on ne peut pas appliquer la formule du binôme. **(1pt)**

Exercice 3

1) $D_1 = 0$ car la première colonne est nulle. **(0,5pt)**
$D_2 = 0$ car la $3^{ème}$ colonne = la $1^{ère}$ colonne + la $2^{ème}$ colonne. **(0,5pt)**
$D_3 = 0$ car la $3^{ème}$ colonne = 2 ·(la $1^{ère}$ colonne). **(0,5pt)**

2) Non **(0,25pt)**
Contre exemple :
$F = IR \times \{0\}, G = \{0\} \times IR$
$(1,0), (0,1) \in F \cup G$ mais $(1,0) + (0,1) = (1,1) \notin F \cup G$. **(0,25pt)**

3) Non **(0,25pt)** : Si la matrice A est de type (n,p), avec $n \neq p$ alors le produit $A^2 = AA$ est impossible car le nombre de colonnes de A est différent du nombre de lignes de A. **(0,25pt)**

4) Non **(0,25pt)** : Une matrice triangulaire n'est pas toujours inversible car il ne faut pas qu'il y'ait de 0 sur sa diagonale principale.
Exemple :
$A = \begin{pmatrix} 1 & 2 \\ 0 & 0 \end{pmatrix}$ $\det A = 0 \Rightarrow A^{-1}$ n'existe pas. **(0,25pt)**

5) Oui, AB est inversible et $(AB)^{-1} = B^{-1}A^{-1}$. **(0,25pt)**
En effet : $(AB)(AB)^{-1} = (AB)(B^{-1}A^{-1}) = A(BB^{-1})A^{-1} = A\,I\,A^{-1} = I$
Et $(AB)^{-1}(AB) = (B^{-1}A^{-1})(AB) = B^{-1}(A^{-1}A)B = B^{-1}\,I\,B = I$. **(0,25pt)**

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de Bourse de formation Post-Graduée à l'Etranger
au titre de l'année universitaire 2009/2010

EPREUVE DE SPECIALITE: Mathématiques et Informatique (MI)
Matière : Algorithmique – Corrigé type sujet n° 2

Exercice 1 (3.75 pts)

```
Elem * FusionListes(Elem * L1, Elem * L2)
début
        Elem * p ;
        si L1 = NIL alors retourner L2 ;
        sinon si L2 = NIL alors retourner L1 ;
                sinon
                si L1.info < L2.info alors
                L := L1 ;
                L1 := L1.next ;
                sinon
                L := L2 ;
                L2 := L2.next ;
                p := L ;
        tant que L1 ≠ NIL et L2 ≠ NIL faire
        si L1.info < L2.info alors
                p.next := L1 ;
                L1 := L1.next ;
                sinon
                p.next := L2 ;
                L2 := L2.next ;
                p := p.next ;
        si L1 ≠ NIL alors p.next := L1 ;
        sinon p.next := L2 ;
        retourner L ;
fin
```


Exercice 2 (3.75 pts)

```
void decompose(int M, int n, int *M1, int *M2)
 {
  int puis10=1;
  int i;
  for (i=1;i<=n;i++) puis10=puis10*10; // calcul de 10^n
  if (puis10>M)
  { //nbr de chiffres de M < n
    *M1=M;
    *M2=0;
  }
  else {
    *M1=M%puis10;
    *M2=M/puis10;
  }
 }
```

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante 3)

CORRIGE.

Exercice 1 : (sur 02,5 pts).

1ère methode : Utilisation de la notion de determinant.

Les vecteurs u_1, u_2 et u_3 sont linéairement dependants si

$$\begin{vmatrix} m & 1 & 1 \\ 1 & m & 1 \\ 1 & 1 & m \end{vmatrix} = (m-1)^2(m+2) = 0 \Leftrightarrow m=-2 \text{ ou } m=1$$

2ème methode : Les vecteurs $\{u_1, u_2, u_3\}$ sont liés s'il existe un triplet $(\alpha, \beta, \gamma) \in \mathbb{R}^3$, non tous nuls tel que $\alpha u_1 + \beta u_2 + \gamma u_3 = 0$ d'ou

$$\begin{cases} m\alpha + \beta + \gamma = 0 \\ \alpha + m\beta + \gamma = 0 \\ \alpha + \beta + m\gamma = 0 \end{cases} \quad (*) \quad 0,5\text{ pt}$$

pour $m = 1$ le système $(*)$ est equivallent à l'equation $\alpha + \beta + \gamma = 0$ qui admet une infinité de solutions. 01pt

pour $m \neq 1$. Par substitution et combinaison des differentes equations du système on obtient :

$$\begin{cases} \alpha m + \beta + \gamma = 0 \\ \alpha - \beta = 0 \\ \alpha - \gamma = 0 \end{cases} \Rightarrow \alpha(m+2) = 0 \quad \text{pour } m=-2$$

01pt il y a une infinité de solutions.

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante3)

Corrigé suite

Exercice 2. 05pts

1. $A(0) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = I$; $A(\alpha) - I = \begin{pmatrix} 0 & 0 & \alpha \\ -\alpha & 0 & -\frac{\alpha^2}{2} \\ 0 & 0 & 0 \end{pmatrix}$

0,5 pt 0,5

2. $A(\alpha) \times A(\beta) = \begin{pmatrix} 1 & 0 & \alpha+\beta \\ -(\alpha+\beta) & 1 & -\frac{(\alpha+\beta)^2}{2} \\ 0 & 0 & 1 \end{pmatrix} = A(\alpha+\beta)$

01 pt

3. D'après 1 et 2, $A(\alpha-\alpha) = A(0) = I = A(\alpha) \cdot A(-\alpha)$
$= A(-\alpha) \cdot A(\alpha)$

D'où A est inversible et $A(-\alpha) = \left(A(\alpha)\right)^{-1}$ 01 pt

4. $b_1 = a_1 + a_2 + a_3$
$b_2 = a_2 + a_3$ $\Rightarrow$ la matrice de passage $P = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$
$b_3 = a_3$

et $P^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}$. 01 pt

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante 3)
CORRIGE

Suite Exo2

Sachant que $M\left(f, (b_j)_{j=\overline{1,3}}\right) = A'(\alpha) = P^{-1} \cdot A(\alpha) \cdot P$

on obtient

$$A'(\alpha) = \begin{pmatrix} 1+\alpha & \alpha & \alpha \\ -2\alpha - \frac{\alpha^2}{2} & 1-\left(\alpha+\frac{\alpha^2}{2}\right) & -\left(\alpha+\frac{\alpha^2}{2}\right) \\ \alpha+\frac{\alpha^2}{2} & \frac{\alpha^2}{2} & 1+\frac{\alpha^2}{2} \end{pmatrix}$$

01 pt

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de bourses de formation post-graduée
A l'étranger au titre de l'année universitaire 2009-2010

CORRIGE DE SPECIALITE – LMD Mathématiques-Informatique

Matière : **Algorithme** (variante N°**3**)

(a) Déclarations

Fichier

```
Const
N = 30;
Type
Piece = record
                Ref : integer;
                Lab : string[N];        0.5
                Pri : real
        End;
Var
F : file of piece;
```

Liste

```
Type
Liste = ^node;
Node = record                           0.5
                Info : piece;
                Suiv : liste
        End;
Var
L : liste;
```

(b) Procedures de rajout

- Au debut

```
Procedure rajoutd(var L:liste; x:piece);
Var
    E:liste;
Begin
      New(E);
      E^.info.ref:=x.ref;
      E^.info.lab:=x.lab;
      E^.info.pri:=x.pri;
      E^.suiv:=L;
      L:=E;
End; {rajoutd}
```

1 pt

- A la fin

```
Procedure rajoutf(var L:liste; x:piece);
Var
    E,pt:liste;
Begin
      New(E);
      E^.info.ref:=x.ref;
      E^.info.lab:=x.lab;
      E^.info.pri:=x.pri;
      Pt:=L;
      While pt^.suiv<>nil do
            Pt:=pt^.suiv;
       E^.suiv:=nil;
       Pt^.suiv:=E
End; {rajoutf}
```

1 pt

- Au milieu à l'adresse adr

```
Procedure rajoutm(var L:liste; adr:liste, x:piece);
Var
    E,pt,prec:liste;
Begin
      New(E);
      E^.info.ref:=x.ref;
      E^.info.lab:=x.lab;
      E^.info.pri:=x.pri;
      Pt:=L;
      Prec:=L;
      While pt<>adr do
            Begin
                  Prec:=pt;
                  Pt:=pt^.suiv
            End
       E^.suiv:=pt;
       Prec^.suiv:=E
End; {rajoutm}
```

1 pt

(c) Procedure de tri

```
Procedure tri(var F : file of piece ; L : liste);
Var
    X : piece;
    E, pt : liste;
Begin
(* initialisation de la liste *)
L := nil;  0.25
assign(F ; 'C:fichier.dat');
Reset(F) ;
(*Insertion des éléments du fichier F dans la liste L*)
While not eof(F) do
        Begin
            Read(F, X);
            New(E);
            E^.info.ref := X.ref;
            E^.info.lab := X.lab;
            E^info.pri := X.pri;
            Pt := L ;
            While (pt <> nil) and (E^.info.ref < pt^.info.ref) do
                Pt := pt^.suiv;
            If pt = L then
              rajoutd(L, E)           (*Le rajout est au début*)
            else
                if pt = nil then
                  rajoutf(L, E)       (*Le rajout est à la fin*)
                else
                  rajoutm(L, E, pt)   (*Le rajout est au milieu à l'@ pt*)
        end;
        (* vider le fichier *)
        Rewrite(F);  0.25
        (*Insertion des noeuds de L dans le fichier F*)
        Pt := L;
        While pt <> nil do
            Begin
                Write(F, pt^.info);
                Pt := pt^.suiv
            End;
        Close(F)
End;(* Tri *)
```

2 pts

1 pt

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante 4)

CORRIGE.

Exercice 1 (sur 2,5 pts)

1. Soit $y \in \mathrm{Im} f \Rightarrow \exists x \in E$ tel que $y = f(x)$

$y = f(x) \Rightarrow f(y) = f(f(x)) = f^2(x) = 0 \Rightarrow y \in \mathrm{Ker} f$.

Donc $\mathrm{Im} f \subset \mathrm{Ker} f$. (01 pt)

2. $\dim E = \dim \mathrm{Im} f + \dim \mathrm{Ker} f = 3$

$\mathrm{Im} f \subset \mathrm{Ker} f \Rightarrow \dim \mathrm{Im} f \leq \dim \mathrm{Ker} f$ (1)

$f \neq 0 \Rightarrow \mathrm{Ker} f \neq E \Rightarrow \dim \mathrm{Ker} f \neq 3$ (2)

(01,5 pts) Comme la dimension est un nombre entier, on a :

$\dim \mathrm{Im} f + \dim \mathrm{Ker} f = 3 = 0+3$ (impossible d'après (2))
$= 3+0$ ou $2+1$ (impossible d'après (1))

Donc le seul cas possible est $\dim \mathrm{Im} f = 1 \wedge \dim \mathrm{Ker} f = 2$

Exercice 2 (5 pts) : $V = \{(x,y,z,t) \in \mathbb{R}^4 / x = y - 3z ; z = 2t\}$

$= \{(y-6t, y, 2t, t) / y,t \in \mathbb{R}\}$

$= \{y(1,1,0,0) + t(-6,0,2,1) / y,t \in \mathbb{R}\}$.

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante 4)

CORRIGE

Suite corrigé Exo2

les vecteurs $k=(1,1,0,0)$ et $\ell=(-6,0,2,1)$ engendrent V

On montre qu'ils sont linéairement indépendants et donc $\{k,\ell\}$ constituent une base de V (01pt)

2. Pour obtenir une base de $\mathbb{R}^4$, il faut trouver 2 vecteurs k' et ℓ' de $\mathbb{R}^4$ tel que le système $\{k,\ell,k',\ell'\}$ soit LIBRE (ou générateur).

Une solution serait de compléter par les éléments de la base canonique de $\mathbb{R}^4$: $k'=e_1=(1,0,0,0)$ et $\ell'=e_3=(0,0,1,0)$ et de vérifier que le système $\{k,\ell,e_1,e_3\}$ est libre. (01pt)

3. $W=\{(x,y,z,t)\in\mathbb{R}^4 \,/\, x-y+2z+t=0\}$

$=\{(x,y,z,t)\in\mathbb{R}^4 \,/\, y=x+2z+t\}$

$=\{(x,x+2z+t,z,t)\in\mathbb{R}^4 \,/\, x,z,t\in\mathbb{R}\}$

$=\{x(1,1,0,0)+z(0,2,1,0)+t(0,1,0,1)\}$

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante 4)

Suite corrigé Exo 2

les vecteurs $v_1 = (1,1,0,0)$; $v_2 = (0,2,1,0)$ et $v_3 = (0,1,0,1)$ engendrent W. On vérifie qu'ils sont libres et Donc ils constituent une base de W. 01 pt

4. $V \cap W = \{(x,y,z,t) \in \mathbb{R}^4 / x = y - 2z,\ z = 2t;\ x = y - 2z - t\}$

$= \{(x,y,z,t) \in \mathbb{R}^4 / x = y;\ t = z = 0\}$

$= \{x(1,1,0,0);\ x \in \mathbb{R}\}$. 02 pts

D'où le vecteur $(1,1,0,0)$ constitue une base de $V \cap W$.

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de bourses de formation post-graduée
A l'étranger au titre de l'année universitaire 2009-2010

CORRIGE DE SPECIALITE – LMD Mathématiques-Informatique

Matière : **Algorithme** (variante N° 4)

1. Déclaration

```
Const
N = 100 ;
M = 25 ;
Type
Element = record
              Info : string[M] ;
              Suiv : integer ;
          End ;
Table = record
            Tete : integer ;
            T : array[1..N] of element ;
        End;
Var
Tab : table;
```


2. Fonction du nombre d'occurrences.

```
Function occur(Tab : table , E : string[M] ): integer;
Var
Co, I : integer;
Begin
With tab do
   Begin
     I := tete;
     Co := 0;
     While I <> -9 do
          Begin
               If T[ I ].Info = E then
                  Co := co + 1;
               I := T[ I].suiv
          End
        End;
Occur := co
End;
```


1,5

3. procédure d'emplacements physique et logique.

```
Procedure emplacement (tab:table; E:string[M]; var ep,el:integer);
Var
      trouve:Boolean;
begin
      trouve:= false;
      with tab do
           begin
                  el:=1;
                  ep:=tete;
                  while (ep<>-9) and (not trouve) do
                         if T[ep].info=E then
                           trouve :=true
                         else
                              begin
                                    ep:=T[ep].suiv;
                                    el:=el+1;
                              end
           end;
          if not trouve then
            writeln('erreur: element non trouvé')
end; {emplacement}
```

2 pts

4. Procédure de rajout dans la table.

```
Procédure rajout (var Tab : table ; E : string[M]) ;
Var
    I, J, prec : integer ;
Begin
    With Tab do
            If  tete = -9 then (* table est vide donc E le 1ier élément à ranger *)
              Begin
                    T[1].info := E ;
                    T[1].suiv := -9 ;        0.5
                    Tete := 1
              End
            Else    (* Table non vide *)
                Begin
                    (* J emplacement physique *)
                    J := 1 ;
                    While (T[J].info <> ' ') and ( J <= N) do        1 pt
                            J := J + 1;
                    If J > N then
                      Writeln(' Erreur : Table pleine ')        0.5
                    Else
                        Begin
                            T[J].Info := E;
                            I := tete;                (* I emplacement logique*)
                            Prec := Tete ;
                            While (I <> -9) and (T[I].Info < E) do
                                Begin
                                  Prec := I;
                                  I := T[I].suiv        1 pt
                                End;
                            T[I].suiv := I;
                            T[prec].suiv := J
                        End
                  End;
    End;
```

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante 5)
CORRIGÉ.

Exercice 1 : 02,5 pt

1. On vérifie que $\forall \alpha, \beta \in \mathbb{R}$, $\forall u_1, u_2 \in \mathbb{R}^3$ on a

$$f(\alpha u_1 + \beta u_2) = \alpha f(u_1) + \beta f(u_2).$$ 0,5 pt

2. $f(e_1) = f(1,0,0) = (2,-1,1)$
$f(e_2) = f(0,1,0) = (-1,1,0)$ D'où
$f(e_3) = f(0,0,1) = (1,0,-1)$

$$M_f = \begin{pmatrix} 2 & -1 & 1 \\ -1 & 1 & 0 \\ 1 & 0 & -1 \end{pmatrix}$$ 01 pt

3. $\ker f = \{ u \in \mathbb{R}^3 \,/\, f(u) = 0_{\mathbb{R}^3} \}$

$= \{ (x,y,z) \in \mathbb{R}^3 \,/\, (2x-y+z, -x+y, x-z) = (0,0,0) \}$.

D'où le système : $\begin{cases} 2x - y + z = 0 \\ -x + y = 0 \\ x - z = 0 \end{cases} \Rightarrow x = y = z = 0$

$\Rightarrow \ker f = \{0_{\mathbb{R}^3}\} \Rightarrow f$ est injective 0,5 pt

P2/3

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante 5)

Suite corrigé Exo 1

$3 = \dim \mathbb{R}^3 = \dim \operatorname{Im} f + \dim \operatorname{Ker} f = 3 + 0$

$\Rightarrow \dim \operatorname{Im} f = 3 \Rightarrow \operatorname{Im} f = \mathbb{R}^3 \Rightarrow f$ est surjective

D'où f est bijective. (0,5 pt)

Exercice 2 (sur 02,5 pt)

$\operatorname{Ker} f = \{ x \in E / f(x) = 0_F \}$.

Comme f est linéaire $\Rightarrow f(0_E) = 0_F \Rightarrow \operatorname{Ker} f \neq \phi$. (0,5)

Soit $u, v \in \operatorname{Ker} f$, soit $\alpha, \beta \in K$ Montrons que

$\alpha u + \beta v \in \operatorname{Ker} f$ ~~a.r.a.~~

$f(\alpha u + \beta v) = \alpha f(u) + \beta f(v)$ (f est linéaire)

$= \alpha \cdot 0 + \beta \cdot 0$ (car $u, v \in \operatorname{Ker} f$)

$= 0$

d'où $\alpha u + \beta v \in \operatorname{Ker} f$. (02 pts)

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Concours national pour l'obtention de bourses de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE-LMD Mathématiques – Informatique
Durée: 45 mn

Matière: Algèbre 2 (Variante 5)

CORRIGÉ

Exercice 3 (sur 2,5 pts)

1 $I + B = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$ 0,5

$B^2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ 0,5

2. $(I+B)^n = ?$

1re methode : raisonnement par reccurence :

2eme methode : Utiliser la formule du Binôme de Newton ($I \cdot B = B \cdot I$ et $B^n = 0 \quad \forall n \geq 2$)

$$(I+B)^n = \sum_{i=0}^{n} C_n^i B^i . I^{n-i}$$ 01,5

$$= C_n^0 \cdot B^0 \cdot I^n + C_n^1 B I^{n-1} + 0$$

$$= I + nB$$

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de bourses de formation post-graduée
A l'étranger au titre de l'année universitaire 2009-2010

CORRIGE DE SPECIALITE – LMD Mathématiques-Informatique

<u>**Matière :**</u> **Algorithme** (variante N° 5)

1. Déclarations

```
Const
      N=26;
Type
     Tab=array['a'..'z'] of integer;          0.5
      Fic:file of char;
Var
    T:tab;
    F:fic;
```

2. sous programme du nombre d'occurrences

```
procedure init(var T:tab);
var
   i: char;
begin
     for i:='a' to 'z' do                     1 pts
       T[i]:=0
end;

procedure compter(var F:fic, T:tab);
var
   x:char;
begin
     reset(F);   0.5
     init(T);
     while not eof(F) do
           begin                              1 pts
                read(F,x) ;
                T[x]:=T[x]+1
           end;   0.5
       close(F)
end;
```

3. affichage du rsultat

```
procedure affiche(T:tab);
var
   i:char;
begin
     i:='a';
     while i<='z' do
          begin
                if T[i]<>0 then
                    writeln(I,'=',T[i]);
                i:=succ(i)
           end
end;
```

2 pts

4. caractere ayant un nombre d'occurrences max

```
function max(T:tab ):char;
var
   m:integer;
    i:char;
begin
     m:=0;
     for i:='a' to 'z' do
          if T[i]>m then
             begin
                   x:=i;
                   m:=T[i]
              end;
       max:=x
end;
```

2 pts

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de bourses de formation
Post-graduée à l'étranger au titre de l'année universitaire 2009-2010

Epreuve de Spécialité .LMD - MI
Matière: Algèbre II – Corrigé de l'épreuve N°6

1°/ Si f est l'endomorphisme de matrice M dans la base canonique de IR^4, l'endomorphisme $g = f^2$ a pour matrice dans la base canonique le carré M^2 de la matrice M.

$$M = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 4 & -6 & 4 & 4 \\ 4 & -4 & 2 & 4 \\ 0 & -4 & 4 & 2 \end{pmatrix} \quad \text{donne} \quad M^2 = \begin{pmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix} = 4I.$$

Cette relation montre que l'on a : $(M - 2I)(M + 2I) = 0$, donc le polynôme minimal de la matrice M est :

$$Pu(X) = (X-2)(X+2)$$ **(1,5points)**

2°/ Les valeurs propres de M^2 sont les racines du polynôme caractéristique $(X-4)^4$ de M^2. Il y en a une seule, 4, d'ordre de multiplicité 4. La matrice M^2 a une seule valeur propre, 4. **(1,5points)**

3°/ Soit λ une valeur propre de M. Pour tout vecteur propre x relativement à cette valeur propre on a $f(x) = \lambda x$, donc $g(x) = f(f(x)) = f(\lambda x) = \lambda f(x) = \lambda^2 x$. Il en résulte que $\mu = \lambda^2$ est valeur propre de g, donc de la matrice M^2 et que x est vecteur propre pour cette valeur propre. **(1,5points)**

4°/ Comme la seule valeur propre de M^2 est 4, les valeurs propres de M vérifient $\lambda^2 = 4$. Les seules valeurs propres possibles de M sont donc -2 et 2.
On a déjà observé que le polynôme minimal de la matrice M est $Pu(X) = (X-2)(X+2)$
Les racines du polynôme minimal de M sont aussi des racines du polynôme caractéristique de M, de sorte que les deux valeurs -2 et 2 sont des racines du polynôme caractéristique de M : ce sont toutes deux des valeurs propres de M. Comme ce sont les seules valeurs possibles pour les valeurs propres, on en conclue : Les valeurs propres de la matrice M sont -2 et 2. **(1,5points)**

5°/ Pour tout vecteur x de E, on a : $(f - 2\,id_E)(f(x) + 2x) = (M - 2I)(M + 2I)x = 0$ et

$$(f + 2\,id_E)(f(x) - 2x) = (M + 2I)(M - 2I)x = 0$$

Puisque $(M - 2I)(M + 2I) = (M + 2I)(M - 2I)x = 0$. D'où les relations :
$f(x) + 2x \in Ker(f - 2\,id_E)$ et $f(x) - 2x \in Ker(f + 2\,id_E)$ **(1,5points)**

République Algérienne Démocratique et Populaire
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
Concours National pour l'obtention de bourse de formation post-graduée
à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE « LMD-Mathématiques et Informatique »

Matière : « Algorithmique » - Sujet N° 6

Corrigés Types (7.5 pts)

<u>Exercice n°1 :</u> (3.5 pts)

```
Algorithme RECDIC
Var Vect : tableau  [1..100] d'entiers ;
      N,val, Pos, Deb,Fin, m  : entier;
      Trouv: booleen;
Debut
Deb:=1; Fin:=N; trouv:= Faux;
Tant que  (Deb < Fin et trouv = Faux) faire
  Debut
     M: = (Deb + Fin)/2;
      Si A[m] < val Alors Deb:= m +1
                     Sinon
                      Si A[m] > val  Alors Fin := m-1
                                  Sinon
                                   Debut
                                      Trouv:= vrai;
                                       Pos:= m;
                                   Fin
    Fin
Si Trouv = vrai alors ecrire ('la valeur est trouvé à la position : ', pos)
Sinon ecrire ('la valeur n'existe pas dans le vecteur')
Fin.
```

<u>Exercice n°2 :</u> (4 pts)

- Ecriture en langage algorithmique :

1. Fonction **FIBREC (** ***n*** **: entier) : entier ;**

```
Début
  si n ≥ 1 alors FIBREC := 1
  sinon FIBREC := FIBREC(n-2)+FIBREC(n-1) ;
  finsi
Fin.
```

2. Fonction FIBIT(*n* : entier) : entier ;
Début
fib := 1; fibA := 1; fibB := 0;
pour i de 1 à n faire
    fib := fibB + fibA ;
    fibB := fibA ;
    fibA := fib ;
finpour
FIBIT :=fib ;
Fin.

- Ecriture en langage C des deux fonctions :

1.
```
int FIBREC ( int n)
{
 if (n <= 1) return 1
else return FINREC(n- 2) + FIBREC(n -1) ;
 }
```

2.
```
int FIBIT ( int n)
{
int fib = 1 , fibA = 1 , fibB = 0 , i ;
for ( i = 1 ; i <= n ; i++) {
                fib = fibB + fibA ;
                fibB = fibA ;
                fibA = fib ;
                    }
return fib ;
}
```


FIN

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de bourses de formation
Post-graduée à l'étranger au titre de l'année universitaire 2009-2010

Epreuve de Spécialité .LMD - MI
Matière: Algèbre II – Corrigé de l'épreuve N°7

1. Soit p un endomorphisme. On a alors:

$$(I-p)o(I-p) = (I-p) + (pop - p)$$

Donc $I-p$ est un projecteur si et seulement si $pop=p$, d'où le résultat.

(1 point)

2. Démontrons d'abord que q est un projecteur :

$$qoq = (p_1 + p_2 - p_2 o\, p_1) o\, (p_1 + p_2 - p_2 o\, p_1)$$

$$= p_1 o\, p_1 + p_1 o\, p_2 - (p_1 o\, p_2) o\, p_1 + p_2 o\, p_1 + p_2 o\, p_2$$

$$- (p_2 o\, p_2) o\, p_1 - p_2 o (p_1 o\, p_1) - p_2 o (p_1 o\, p_2) + p_2 o (p_1 o\, p_2) o\, p_1$$

$$= p_1 + p_2 o\, p_1 + p_2 - p_2 o\, p_1 - p_2 o\, p_1 = q.$$ **(1,5points)**

Il est clair que si, $x \in \operatorname{Ker} p_1 \cap \operatorname{Ker} p_2$ alors :

$$q(x) = p_1(x) + p_2(x) - p_2(p_1(x)) = 0$$

donc $x \in \operatorname{Ker} q$ et $\operatorname{Ker} p_1 \cap \operatorname{Ker} p_2$ est inclus dans $\operatorname{Ker} q$. **(1 point)**

Soit $x \in \operatorname{Ker} q$ on a :

$$p_1(x) + p_2(x) - p_2(p_1(x)) = 0 \quad (1).$$

En appliquant p_1 aux deux membres de cette égalité , on trouve :

$$p_1(p_1(x)) + p_1(p_2(x)) - p_1(p_2(p_1(x))) = p_1(x) = 0$$

donc $x \in \operatorname{Ker} p_1$. En faisant alors $p_1(x)=0$ dans l'égalité (1), on trouve $p_2(x)=0$, donc $x \in \operatorname{Ker} p_2$ et par suite Ker q est inclus dans $\operatorname{Ker} p_1 \cap \operatorname{Ker} p_2$. **(1,5points)**

Soit $y \in \operatorname{Im} q$, il existe $x \in E$ tel que :

$$y = q(x) = p_1(x) + p_2(x - p_1(x))$$

ce qui prouve que $y \in \operatorname{Im} p_1 + \operatorname{Im} p_2$ et que Im q est inclus dans $\operatorname{Im} p_1 + \operatorname{Im} p_2$. **(1 point)**

Soit $y \in \operatorname{Im} p_1 + \operatorname{Im} p_2$, Il existe x_1 , $x_2 \in$ E tels que $y = p_1(x) + p_2(x)$. On remarque alors :

$$p_1(y) = p_1(p_1(x_1)) + p_1(p_2(x_2)) = p_1(x_1)$$

$$p_2(y) = p_2(p_1(x_1)) + p_2(p_2(x_2)) = p_2(p_1(x_1)) + p_2(x_2)$$

On a :

$$p_2(p_1(y)) = p_2(p_1(x_1)).$$

En comparant ces égalités on obtient

$$y = p_1(y) + p_2(y) - p_2(p_1(y)) = q(y)$$

ce qui prouve que $y \in \operatorname{Im} q$ et par suite que $\operatorname{Im} p_1 + \operatorname{Im} p_2$ est inclus dans Im q. **(1,5points)**

République Algérienne Démocratique et Populaire
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
Concours National pour l'obtention de bourse de formation post-graduée à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE « LMD-Mathématiques et Informatique »

Matière : « Algorithmique » - Sujet N°7

Corrigés Types (7.5 pts)

<u>Exercice n°1 :</u> (3.5 pts)

1.
```
Algorithme TRIBUL ;
Var
    V : tableau[1..100] d'entiers ;

Début
    droite, i, z : entier ;
    inversion : booleen
    droite ← n
    Inversion ← vrai ;
        Tantque (inversion) faire
            inversion := faux
            pour i := 1 à (droite-1) faire
                si (T(i)> T(i+1) ) alors Z := T(i) ; T(i) := T(i+1) ; T(i+1) := z ;
                inversion ← Vrai ;
                fin si
            fin pour ;
    droite ← droite – 1 ;
    Fintantque
Fin
```

2. Considérons le nombre d'opérations du tri bulle, pour un tableau n éléments. Dans le pire des cas, ce qui correspond à un tableau préalablement trié de manière décroissante, il a des permutations jusqu'à ce que la liste ait été parcourue (n-1) fois.
Dans le cas du tri bulle classique, cela revient à faire (n-1) fois (n-1) comparaisons (et quelques permutations). Soit une complexité proportionnelle à $(n-1)^2=n^2-2n+1$, soit en $O(n^2)$.

Exercice n°2 : (4 pts)

```
Program Ouvrage
Type
 Date = record
  Jour : 1..31 ;
  Mois : 1..12 ;
  Année : 1950...2007 ;
 End ;
     Ouv = record
      COTE : integer ;
      TITRE : String
      AUTEUR : String ;
       EDITEUR:  String;
      NB_EXP : integer ;
      DAT_PAR :  Date
  End ;
Var
  F : File of Ouv;
  O : Ouv;
  I : Integer;

Procedure CREAT;
 Debut
   Rewrite(F)
  End.

Procedure SAISIE;
 Begin
I := 1;
While I <= 1000  Do
 Begin
  Clear scr ;
   Writeln('Cote'); Readln(O.COTE);
   Writeln('Titre'); Readln(O.TITRE);
   Writeln('Auteur'); Readln(O.AUTEUR);
   Writeln('Editeur'); Readln(O.EDITEUR);
   Writeln('Nombre d"exemplaire'); Readln(O.NB_EXP);
   Writeln('Date de Parution'); Readln(O.DAT_PAR);
   Write(F,O);
   I:= I+1
 End;
  Close(F)
 End.
```

```
Procedure AFFICH;
Begin
 Assign (F,'AA.DAT');
RESET (F); Read (F,O);
While Not EOF(F) Do
   Begin
     if (O.Année = 2005) then
       Begin
          Write (O.COTE, O.TITRE, O.AUTEUR, …., O. DAT_PAR);
          Read(F, O);
       End;
    End;
   Close(F);
End.

Begin
  CREAT;
  SAISIE;
  AFFICH;
End.
```


FIN

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de bourses de formation
Post-graduée à l'étranger au titre de l'année universitaire 2009-2010

Epreuve de Spécialité .LMD - MI
Matière: Algèbre II – Corrigé de l'épreuve N°8

01. Faux.

Soit f un endomorphisme de E vérifiant $f^2 = f$. On a alors $Dét\ (f^2) = Dét\ (f)$, soit : $(Dét\ (f))^2 = Dét\ (f)$. d'où $Dét\ (f) = 0$ ou $Dét\ (f) = 1$.
Si $Dét\ (f) = 1$, f est inversible, mais si $Dét\ (f) = 0$, f n'est pas inversible.
Par exemple, si l'on prend pour f le projecteur sur le premier axe canonique :
$f : (x_1, x_2, \dots, x_n) = (x_1, 0, \dots, 0)$
on a bien :
$f^2\ (x_1, x_2, \dots, x_n) = f(f(x_1, x_2, \dots, x_n)) = f(x_1, 0, \dots, 0) = (x_1, 0, \dots, 0)$
$= f(x_1, x_2, \dots, x_n)$
donc $f^2 = f$, mais le noyau de f est l'ensemble des vecteurs dont la première composante est nulle : c'est un sous-espace vectoriel de dimension $n - 1$: il n'est pas réduit à 0 dès que n est strictement plus grand que 1. Comme le noyau n'est pas réduit à 0, f n'est pas inversible. **(1,5points)**

02. Faux.

L'endomorphisme inverse f^{-1} possède les mêmes sous-espaces propres que f avec pour valeurs propres les inverses des valeurs propres de f, ce qui résulte de :

$$f(x) = \lambda x \iff f^{-1}(x) = \lambda^{-1} x$$

Dire que E possède une base formée de vecteurs propres pour f est donc équivalent à dire que E possède une base formée de vecteurs propres pour f^{-1}. Donc, pour un endomorphisme inversible, f^{-1} est diagonalisable si et seulement si, f est diagonalisable.
Or il existe des endomorphismes inversibles non diagonalisables, par exemple une rotation d'angle θ dans IR^2. **(1,5points)**

03. Vrai. $f(x) = \lambda x \Rightarrow f^2(x) = f(\lambda x) = \lambda f(x) = \lambda^2 x$

Tout vecteur propre pour f est vecteur propre pour f^2. Donc s'il existe une base formée de vecteurs propres pour f, cette base est formée de vecteurs propres pour f^2. Il en résulte que : f diagonalisable $\Rightarrow f^2$ diagonalisable. **(1,5points)**

04. Vrai.

Un endomorphisme idempotent est, par définition, un projecteur. Tout projecteur f est diagonalisable car E est somme directe du noyau et de l'image du projecteur, et, dans le noyau, f est nul, et, dans l'image, f est l'application identique, donc :

$$f^2 = f \Rightarrow f \text{ est diagonalisable}$$ **(1,5points)**

05. Vrai

On a pour un endomorphisme inversible, f^{-1} est diagonalisable si et seulement si, f est diagonalisable. Donc :

f inversible et diagonalisable $\Rightarrow f^{-1}$ diagonalisable **(1,5points)**

République Algérienne Démocratique et Populaire
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
Concours National pour l'obtention de bourse de formation post-graduée à l'étranger au titre de l'année universitaire 2009-2010

EPREUVE DE SPECIALITE « LMD-Mathématiques et Informatique »

Matière : « Algorithmique » - Sujet N° 8

Corrigés Types (7.5 pts)

<u>Exercice n°1 :</u> (3.5 pts)

```
TYPE
liste=record
     val :Integer ;
     suiv :^liste ;
End ;

Var  tête :^liste ;
   Begin
     P := tête ;
     If p<>NIL Then
                    Begin
                       Vmax := P^.val; Vmin:= P^.val;
                       P:= P^.suiv;
                       While (P<> NIL) do
                          Begin
                          If (P^.val > Vmax) Then Vmax:= P^.val
                                                  Else if ( P^.val < Vmin) Then Vmin:= P^.val;
                          P:= P^.suiv;
                          End;
                    Writeln(Vmax,Vmin);
                    End;
    Else Writeln("la liste est vide");
   End.
```

Exercice n°2 : (4 pts)

```
Algorithme rech_pos ;

Var  POS, LONG,i, Lmax : entier ;
       Suite : booléen ;
       T :tableau[1..100] :entier ;
Début
POS := -1 ;
Lmax := 0 ;
Suite := faux ;
Lire(N) ;
Pour i allant de 1 à N faire
Si ( T[i] = 0 ) alors début
                    Si ( non suite ) alors début
                                            LONG := 0 ;
                                            Suite := vrai ;
                                            Fin ;
                     LONG := LONG+1 ;
                     FinSi ;
Sinon Si ( Suite ) alors  début
                          Suite := faux ;
                          Si (LONG > Lmax) alors début
                                                     Lmax := LONG ;
                                                     POS  :=  i -LONG ;
                                                     Fin ;
                          FinSi ;
                          Si (( Suite ) et ( LONG >Lmax )) alors POS := i – LONG + 1 ;
        FinSi ;
Ecrire(POS,LONG) ;
Fin.
```

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de Bourse de formation Post-Graduée à l'Etranger
au titre de l'année universitaire 2009/2010
EPREUVE DE SPECIALITE : Mathématiques et Informatique(MI)

Matière : Algèbre 2 - Corrigé type sujet n°3
Durée : 45 mn

Exercice 1

1) Soit $(x,y,z) \in IR^3$,
$(f\circ f\circ f)(x,y,z) = (f\circ f)(f(x,y,z)) = (f\circ f)(y,z,0) = f(f(y,z,0)) = f(z,0,0) = (0,0,0)$.
D'où $f\circ f\circ f$ est l'application nulle de IR^3. **(1pt)**

2) Comme $Card\, B = 3 = \dim IR^3$, il suffit de montrer que B est libre.

$$\det(v_0, f(v_0), (f\circ f)(v_0)) = \begin{vmatrix} x_0 & y_0 & z_0 \\ y_0 & z_0 & 0 \\ z_0 & 0 & 0 \end{vmatrix} = -z_0^3 \neq 0$$

Donc $\{v_0, f(v_0), (f\circ f)(v_0)\}$ est une famille libre. **(1pt)**

3) $T = M(f,B,B) = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$ **(0,5pt)**

T^3 est la matrice nulle car T^3 est la matrice de $f\circ f\circ f$ dans la base B. **(0,5pt)**
Ou bien :

$T^2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$ **(0,25pt)** $T^3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ **(0,25pt)**

4)

$(2I_3 + T) = \begin{pmatrix} 2 & 0 & 0 \\ 1 & 2 & 0 \\ 0 & 1 & 2 \end{pmatrix}$ $(2I_3 + T)^2 = \begin{pmatrix} 4 & 0 & 0 \\ 4 & 4 & 0 \\ 1 & 4 & 4 \end{pmatrix}$ $(2I_3 + T)^3 = \begin{pmatrix} 8 & 0 & 0 \\ 12 & 8 & 0 \\ 6 & 12 & 8 \end{pmatrix}$

Pour $n \geq 4$:
$(2I_3 + T)^n = \sum_{k=0}^{n} C_n^k (2I_3)^{n-k} T^k = C_n^0 (2I_3)^n + C_n^1 (2I_3)^{n-1} T + C_n^2 (2I_3)^{n-2} T^2$. **(0,5pt)**

Exercice 2

1) Fausse **(0,25pt)** : pour $(x,y) = (5,2) \in A$ et $\lambda = -1 \in IR$, $\lambda(x,y) = (-5,-2) \notin A$. **(0,25pt)**
2) Fausse **(0,25pt)** : car $(5,5),(2,-2) \in B$ mais $(5,5)+(2,-2) = (7,3) \notin B$. **(0,25pt)**
3) Fausse **(0,25pt)** : car $Card\, C = 2 < \dim IR^3 = 3$. **(0,25pt)**
4) Fausse **(0,25pt)** : car $Ker\, g = \{z \in \mathbb{C}, g(z) = 0\} = \{z \in \mathbb{C}, z = \bar{z}\} = IR \neq \{0\}$. **(0,25pt)**
5) Fausse **(0,25pt)** : car $\underbrace{\dim IR_{n+1}[X]}_{n+2} > \underbrace{\dim IR_n[X]}_{n+1}$. **(0,25pt)**
6) Fausse **(0,25pt)** : pour $A = I$ et $B = -I$, la matrice $A + B = 0$ n'est pas inversible. **(0,25pt)**
7) Vraie **(0,25pt)** : c'est le sous-espace vectoriel de dimension minimale
que l'on peut construire, on convient qu'il est de dimension 0. **(0,25pt)**
8) Fausse **(0,25pt)** : car $Card\, B = 3 > \dim IR^2 = 2$. **(0,25pt)**

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Concours National pour l'obtention de Bourse de formation Post-Graduée à l'Etranger
au titre de l'année universitaire 2009/2010

EPREUVE DE SPECIALITE: Mathématiques et Informatique (MI)
Matière : Algorithmique – Corrigé type sujet n°9

Exercice 1 (3.75pts)

On voudrait éliminer les doubles d'un vecteur d'entiers : Proposez un sous-programme qui supprime les doubles sans utiliser un autre vecteur. Le vecteur sera envoyé comme paramètre du sous programme.

Hypothèse : la première case du tableau est d'indice 1. Le tableau est non trié.

```
Procedure ElimineDoubles(var t :tableau, var n :entier) ;
Variables
        Existe : Booléen ;
Début
        i:=2 ;
        tant que i <=n  (Au lieu du for dans le corrigé-type initial)
        faire
                //rechercher si l'élément existe dans la séquence 1 à i-1
                //un booléen existe permet de sortir de la boucle dans l'affirmative
                existe := faux ; j :=1 ;
                Tant que non existe et j<i
                Faire
                        Si t[i] = t[j] alors
                                existe = vrai
                        Sinon
                                j = j+1 ;
                        fsi ;
                fait
                //si l'élément existe déjà, il s'agit de remplacer l'occurrence courante (t[i])
                //par le dernier élément du tableau
                Si existe alors
                        t[i] := t[n] ;
                        n := n-1 ;
                sinon
                        i :=i+1 ;          (le sinon induit par le tant que au lieu du for)
                fsi
        fait
fin
```

Exercice 2(3.75pts)

Ecrire une fonction récursive (en Pascal ou C) qui trouve le maximum des valeurs d'un vecteur d'entiers.

L'idée consiste à se dire : le maximum des valeurs d'un vecteur, c'est la plus grande d'entre deux valeurs : le maximum des valeurs de la première moitié du vecteur et le maximum des valeurs de la seconde moitié

```
int maximum(int t[], int BorneInf, int BorneSup)
{
        int Sortie ;
        if (BorneInf==BorneSup) //Il n'y a plus qu'un seul élément à traiter
                                //c'est donc lui le max
```

```
                Sortie = t[BorneInf] ;
        else
                if (BorneInf == BorneSup-1)        //deux elements restants
                Sortie = max(t[BorneInf],t[BorneSup])
                else
                {
                        Sortie = max(maximum(t, BorneInf, (BorneInf+BorneSup)/2),
                                    maximum(t, (BorneInf+BorneSup)/2+1, BorneSup)) ;
                }
        return Sortie ;
}

int max(int a, int b)
{
        int sortie = a ;
        if (b>a) sortie = b ;
        return sortie ;
}
```