

Corrigé de la série d'exercice n04

Exercice 1 :

```
Algorithme S4_EXO1
Var
T:Tableau[0..99] entier
n,i: entier
debut
  pour i=0 à 99 faire
    tab[i] ← i;
  pour i=0 à 99 faire
    ecrire(tab[i], " ")
  finpour
fin
```

```
#include <stdio.h>
int main() {
  int tab[100],i;
  for( i = 0 ; i < 100 ; i++) tab[i] = i;
  for( i = 0 ; i < 100 ; i++)
    if((i+1) % 10 == 0)
      printf("%.2d\n", tab[i]);
    else
      printf("%.2d,", tab[i]);
  return 0;
}
```

Exercice 2 :

```
Algorithme S4_EXO2
Var
T:Tableau[0..99] entier
n,i: entier
debut
  ecrire("introduire n")
  lire(n)
  pour i=0 à n-1 faire
    ecrire("tab[" ,i,"]=")
    lire(tab[i])
  finpour
  pour i=0 à n-1 faire
    si(tab[i] mod 2 =0)
      ecrire(tab[i], " ")
    fisi
  finpour
  pour i=0 à n-1 faire
    si(tab[i] mod 2 =1)
      ecrire(tab[i], " ")
    fisi
  finpour
fin
```

```
#include <stdio.h>
int main() {
  int tab[100],i,n;
  printf("introduire n ");
  scanf("%d",&n);
  for( i = 0 ; i < n ; i++){
    printf("tab[%d]=",i);
    scanf("%d",&tab[i]);
  }
  printf("Nombres pairs: \n");
  for( i = 0 ; i < n ; i++)
    if(tab[i] % 2 == 0)
      printf("%.2d\t", tab[i]);
  printf("\nNombres impairs: \n");
  for( i = 0 ; i < n ; i++)
    if(tab[i] % 2 != 0)
      printf("%.2d\t", tab[i]);
  return 0;
}
```

Corrigé de la série d'exercice n04

Exercice 5 :

- 1- Même nombre de lignes et de colonnes pour les deux matrices.
- 2- Le nombre de colonnes pour la première matrice est égal au nombre de colonnes pour la deuxième matrice.
- 3- Pour pouvoir calculer la somme et le produit de deux matrices M1 et M2, il faut être des matrices carrées.

Algorithme S4_EXO5

Var

M1,M2,S,P:Tableau[0..99][0..99] entier

n,i,j,k: entier

debut

ecrire("introduire n")

lire(n)

pour i=0 à n-1 **faire**

pour j=0 à n-1 **faire**

ecrire("M1[" ,i,"][",j,"]="")

lire(M1[i][j])

finpour

finpour

pour i=0 à n-1 **faire**

pour j=0 à n-1 **faire**

ecrire("M2[" ,i,"][",j,"]="")

lire(M2[i][j])

finpour

finpour

pour i=0 à n-1 **faire**

pour j=0 à n-1 **faire**

$S[i][j] \leftarrow M1[i][j] + M2[i][j]$

finpour

finpour }

pour i=0 à n-1 **faire**

pour j=0 à n-1 **faire**

pour k=0 à n-1 **faire**

$S[i][j] \leftarrow S[i][j] + M1[i][k] * M2[k][j]$

Finpour

finpour

finpour

pour i=0 à n-1 **faire**

pour j=0 à n-1 **faire**

ecrire(S[i][j], " ")

finpour

finpour

pour i=0 à n-1 **faire**

pour j=0 à n-1 **faire**

ecrire(P[i][j], " ")

finpour

finpour

fin

Corrigé de la série d'exercice n04

<pre>#include <stdio.h> int main() { int l[10][10],M2[10][10],i,j,k,S[10][10],P[10][10],n; printf("entrer n :"); scanf("%d",&n); for (i=0;i<n;i++) for (j=0;j<n;j++){ printf("M1[%d][%d]=",i,j); scanf("%d",&M1[i][j]); } for (i=0;i<n;i++) for (j=0;j<n;j++){ printf("M2[%d][%d]=",i,j); scanf("%d",&M2[i][j]); } for (i=0;i<n;i++) for (j=0;j<n;j++){ S[i][j]=0; P[i][j]=0; } for (i=0;i<n;i++) for (j=0;j<n;j++){ S[i][j]=M1[i][j]+M2[i][j]; } for (i=0;i<n;i++) for (j=0;j<n;j++) for (k=0;k<n;k++){ P[i][j]+=M1[i][k]*M2[k][j]; } }</pre>	<pre> printf("\nM1 :\n"); for (i=0;i<n;i++){ for (j=0;j<n;j++) printf("%.2d\t",M1[i][j]); printf("\n"); } printf("\nM2 :\n"); for (i=0;i<n;i++){ for (j=0;j<n;j++) printf("%.2d\t",M2[i][j]); printf("\n"); } printf("\nS :\n"); for (i=0;i<n;i++){ for (j=0;j<n;j++) printf("%.2d\t",S[i][j]); printf("\n"); } printf("\nP :\n"); for (i=0;i<n;i++){ for (j=0;j<n;j++) printf("%.2d\t",P[i][j]); printf("\n"); } } return 0;}</pre>
---	---

Corrigé de la série d'exercice n04

Exercice 6 :

Algorithme S4_EXO5

Var

M:Tableau[0..99][0..99] entier

S:Tableau[0..99] entier

i,j,n,m,max,J_max,min,J_min,temp: entier

debut

ecrire("introduire n et m")

lire(n,m)

pour i=0 à n-1 **faire**

pour j=0 à m-1 **faire**

ecrire("M[" ,i,"][" ,j,"]="")

lire(M[i][j])

finpour

finpour

pour j=0 à m-1 **faire**

 S[j]←0

pour i=0 à n-1 **faire**

 S[j]← S[j]+ M[i][j]

finpour

finpour

ecrire("M avant permut=")

pour i=0 à n-1 **faire**

pour j=0 à m-1 **faire**

ecrire(S[i][j], " ")

finpour

finpour

ecrire("S=")

pour i=0 à m-1 **faire**

ecrire(S[i], " ")

finpour

max←min←S[0]

J_max←J_min←0

pour i=0 à m-1 **faire**

si(max<S[i])**alors**

 max←S[i]

 J_max←i

sinon

si(min>S[i])**alors**

 min←S[i]

 J_min←i

Finsi

Finsi

finpour

ecrire("Val Max : S[" ,J_max,"]="",max)

ecrire("Val Min : S[" ,J_min,"]="",min)

pour i=0 à n-1 **faire**

 temp←M[i][J_max]

 M[i][J_max] ←M[i][J_min]

 M[i][J_min] ←temp

finpour

ecrire("M Après permutation :")

pour i=0 à n-1 **faire**

pour i=0 à m-1 **faire**

ecrire(" ",M[i][j])

finpour

finpour

fin

Corrigé de la série d'exercice n04

<pre>#include <stdio.h> int main() { int M[10][10],S[10],i,j,n,m,max,J_max,min,J_min,temp; printf("entrer n et m :"); scanf("%d%d",&n,&m); for (i=0;i<n;i++) for (j=0;j<m;j++){ printf("M[%d][%d]=",i,j); scanf("%d",&M[i][j]); } for (j=0;j<m;j++){ S[j]=0; for (i=0;i<n;i++) S[j]+=M[i][j]; } printf("\nM avant permutation :\n"); for (i=0;i<n;i++){ for (j=0;j<m;j++) printf("%.2d\t",M[i][j]); printf("\n"); } printf("\nSomme_Colonne :\n"); for (i=0;i<m;i++){ printf("%.2d\t",S[i]); } }</pre>	<pre>//Calcule J_max et J_min max=min=S[0]; J_max=J_min=0; for (i=1;i<m;i++) if(max<S[i]){ max=S[i]; J_max=i; }else if(min>S[i]){ min=S[i]; J_min=i; } printf("\nVal Max : S[%d]=%d\n",J_max,max); printf("\nVal Min : S[%d]=%d\n",J_min,min); // permutation for (i=0;i<n;i++){ temp=M[i][J_max]; M[i][J_max]=M[i][J_min]; M[i][J_min]=temp; } printf("\nM Après permutation :\n"); for (i=0;i<n;i++){ for (j=0;j<m;j++) printf("%.2d\t",M[i][j]); printf("\n"); } return 0;}</pre>
--	---

Corrigé de la série d'exercice n04

Exercice 7:

<pre>Algorithme S4_EXO1 Var mot:Tableau[0..49] caractère test,i: entier debut ecrire("entrer un mot") lire(mot) test←1 i←0 TQ i<longueur(mot)/2 et test=1 faire si(mot[i]<> mot[longueur(mot)-1-i])alors test←0 finsi i←i+1 finTQ fin si(test=1)alors écrire("Le mot entré est palindrome") sinon écrire("Le mot entré n'est pas palindrome") finsi fin</pre>	<pre>#include <stdio.h> #include <string.h> int main() { char mot[50]; int i=0,test=1; printf("entrer un mot\n"); gets(mot); while(i<strlen(mot)/2 && test==1){ if(mot[i]!=mot[strlen(mot)-1-i]) test=0; i++; } if(test==1) printf("\nLe mot entré est palindrome\n"); else printf("\nLe mot entré n'est pas palindrome\n"); return 0; }</pre>
--	---

Corrigé de la série d'exercice n04

Exercice 8:

Algorithme S4_EXO8

Var

ch1,ch2:Tableau[0..49] caractère

co,i,j,test: entier

debut

ecrire("entrer Ch1")

lire(ch1)

ecrire("entrer Ch2")

lire(ch2)

pour i=0 à longueur(ch1)-1 faire

test←1

j←0

TQ j<longueur(ch2) et test=1**faire**

si(ch1[i+j]<> ch2[j])**alors**

test←0

finsi

j←j+1

finTQ

si(test=1)**alors**

co←co+1

finsi

finpour

ecrire("ch2 existe", co," fois dans ch1")

fin

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main() {
```

```
    char ch1[50],ch2[50];
```

```
    int co=0,i,j,test;
```

```
    printf("entrer Ch1\n");
```

```
    scanf("%s",ch1);
```

```
    getchar();
```

```
    printf("entrer Ch2\n");
```

```
    scanf("%s",ch2);
```

```
    getchar();
```

```
    for(i=0;i<strlen(ch1);i++){
```

```
        test=1;
```

```
        j=0;
```

```
        while(j<strlen(ch2) && test==1){
```

```
            if(ch1[i+j]!=ch2[j])
```

```
                test=0;
```

```
                j++;
```

```
            }
```

```
            if(test==1)
```

```
                co++;
```

```
        }
```

```
    printf("ch2 existe %d fois dans ch1 ",co);
```

```
    return 0;
```

```
}
```