

Solution de l'examen ASD1 – 2026

Exercice 1 : (5points)

Algorithme Exercice1

Variables

montantCommande : réel;
estSaida : chaîne;
fraisLivraison : entier;
majoration : entier;
total : réel;

(1 pt)

Début

// Demande du montant de la commande (0,5 pt)

Écrire("Entrez le montant de la commande (en DA) : ")
Lire(montantCommande)

// Demande si c'est une livraison à Saida (0,5 pt)

Écrire("Livraison à Saida ? (oui/non) : ")
Lire(estSaida)

// Initialisation des frais de livraison (1 pt)

Si (montantCommande > 3000)Alors
 fraisLivraison ← 0
Sinon
 Si (montantCommande >= 1000 ET montantCommande <= 3000) Alors
 fraisLivraison ← 200
 Sinon
 fraisLivraison ← 500
FinSi

// Majoration pour hors Saida (0,5 pt)

majoration ← 0
Si (estSaida = "non") Alors
 majoration ← 600
FinSi

// Calcul du total (0,5 pt)

total ← montantCommande + fraisLivraison + majoration

// Affichage du résultat (1 pt)

Écrire (" TOTAL À PAYER : ", total, " DA")

Fin

Exercice 2 : (6 points)

Algorithme Exercice2

Variables

Mat [20][20] : entier ;
T [20] réel ;
n : entier ;
i, j : entiers ;
moyDiag : réel ;
SDiag ,Occ : entier ;
sommeLigne : entier

(1 pt)

Début

// 1. Saisie des éléments de la matrice (1pt)

Répéter

Écrire("Entrez la taille de la matrice carrée ($n \leq 20$) : ")

Lire(n)

Jusqu'à ($n > 0$ ET $n \leq 20$)

Écrire("Saisie des éléments de la matrice :")

Pour i de 1 à n faire

Pour j de 1 à n faire

Écrire("Mat[", i, ",", j, "] = ") ;

Lire(Mat[i][j]);

FinPour

FinPour

// 2. Calcul de la moyenne de la diagonale principale (1pt)

SDiag $\leftarrow$ 0 ;

Pour i de 1 à n faire

SDiag $\leftarrow$ SDiag + Mat[i][i] ;

FinPour

moyDiag $\leftarrow$ SDiag / n;

Écrire("Moyenne de la diagonale principale = ", moyDiag);

// 3. Création du tableau T avec la moyenne de chaque ligne (1pt)

Pour i de 1 à n faire

sommeLigne $\leftarrow$ 0 ;

Pour j de 1 à n faire

sommeLigne $\leftarrow$ sommeLigne + Mat[i][j] ;

FinPour

T[i] $\leftarrow$ sommeLigne / n ;

FinPour

Écrire("Voici les éléments du tableau :") ;

Pour i de 1 à n faire

Écrire("T[", i, "] = ", T[i]) ;

FinPour

// 4. Affichage des éléments impairs de la matrice (1pt)

```

Écrire("Éléments impairs de la matrice :")
Pour i de 1 à n faire
    Pour j de 1 à n faire
        Si (Mat[i][j] mod 2 <> 0) Alors
            Écrire("Mat[" , i , " , " , j , " ] = " , Mat[i][j] ) ;
        FinSi
    FinPour
FinPour

```

// 5. Nombre d'occurrences de l'élément 4 dans le tableau T (1pt)

```

Occ ← 0
Pour i de 1 à n faire
    Si T[i] = 4 Alors
        Occ ← Occ + 1 ;
    FinSi
FinPour
Écrire("Nombre d'occurrences de 4 dans T : " , Occ) ;

```

Fin

Exercice 3 : (4 points)

Algorithme Exercice3

```

Type Employe = Structure {
    Nom : chaîne ;
    DN : Date ;
    Fonction : chaîne ;
    Salaire : réel ; }
Type Date = Structure {
    j, m, a : entier ;
}

```

(1 pt)

Variables

```

T[100] : Employe ;
n : entier
i : entier
masseSalariale, salaireMax : réel;
indiceMax : entier;

```

Début

```

// Saisie des informations de n employés (1pt)
Répéter
    Écrire("Entrez le nombre d'employés (n ≤ 100) : ")
    Lire(n)
Jusqu'à (n > 0 ET n ≤ 100)

```

```
Écrire("Saisie des informations des employés :")
```

```
Pour i de 1 à n faire
```

```
    Écrire("Nom : ") ;
```

```
    Lire(T[i].Nom) ;
```

```
    Écrire("Date de naissance : ") ;
```

```
    Lire(T[i].DN.j , T[i].DN.m , T[i].DN.a ) ;
```

```
    Écrire("Fonction : ") ;
```

```
    Lire(T[i].Fonction) ;
```

```
    Écrire("Salaire : ") ;
```

```
    Lire(T[i].Salaire) ;
```

```
FinPour
```

```
// Affichage des employés avec salaire > 30000 DA (0,5 pt)
```

```
Écrire("Employés avec un salaire supérieur à 30000 DA :") ;
```

```
Pour i de 1 à n faire
```

```
    Si T[i].Salaire > 30000 Alors
```

```
        Écrire(T[i].Nom , " ") ;
```

```
    FinSi
```

```
FinPour
```

```
// Recherche de l'employé avec le plus grand salaire (1 pt)
```

```
indiceMax ← 1 ;
```

```
salaireMax ← T[1].Salaire
```

```
Pour i de 2 à n faire
```

```
    Si (T[i].Salaire > salaireMax) Alors
```

```
        salaireMax ← T[i].Salaire ;
```

```
        indiceMax ← i ;
```

```
    FinSi
```

```
FinPour
```

```
Écrire("Employé avec le plus grand salaire : " , T[indiceMax].Nom) ;
```

```
// Calcul et affichage de la masse salariale ( 0,5 pt)
```

```
masseSalariale ← 0 ;
```

```
Pour i ← 1 à n faire
```

```
    masseSalariale ← masseSalariale + T[i].Salaire ;
```

```
FinPour
```

```
Écrire("Masse salariale totale : " , masseSalariale , " DA") ;
```

```
Fin
```

Exercice 4 : (5 points)

Erreur dans la ligne N°	Correction de la ligne
[3]	int i, T[10], n, MAX ; 0,5 pt
[6] 0,25 pt	scanf ("%d ",&n); 0,5 pt
[7] 0,25 pt	}while(n<=0 n>=10); 0,5 pt
[8] 0,25 pt	for (i=0 ; i < n ; i++) { 0,5 pt
[12] 0,25 pt	MAX = T[0] ; 0,5 pt
[15] 0,25 pt	MAX = T[i] ; 0,5 pt
[18] 0,25 pt	printf("Voici la valeur maximale : %d", MAX) ; 0,5 pt