

EXAMEN DE SYNTHÈSE 2 D'INFORMATIQUE

Partie I : Questions de cours (2pts)

- 1) Quelle est la différence entre la boucle *tant que* et la boucle *répéter* ?
- 2) A quoi servent les fonctions *round*, *ord* et *chr* ?
- 3) Définissez la notion de *procédure* en algorithmique et donnez un exemple ?
- 4) Que représente une *matrice* en algorithmique ?

Partie II : Exercices

Exercice 1 : Dérouler ces deux algorithmes (1pt+1pt)

<p>Algorithme dérouler 1 Type Tab=Tableau [2, 2] en Entier Variables i, j, val en Entier X : Tab</p> <p>Début val ← 1 Pour i de 1 à 2 faire Pour j de 1 à 2 faire X[i, j] ← Val Val ← Val + 1 Fin Pour Fin Pour Pour i de 1 à 2 faire Pour j de 1 à 2 faire Ecrire (X[i, j]) Fin Pour Fin Pour Fin</p>	<p>Algorithme dérouler 2 Type Tab=Tableau [3, 2] en Entier Variables k, m, en Entier T :Tab</p> <p>Début Pour k de 1 à 3 faire Pour m de 1 à 2 faire T[k, m] ← (k + 1) + 4 * m Fin Pour Fin Pour Pour k de 1 à 3 faire Pour m de 1 à 2 faire Ecrire (T[k, m]) Fin Pour Fin Pour Fin</p>
---	--

Exercice 2 (3 pts)

Ecrire un *algorithme* qui permet de lire une suite de caractères se terminant par un espace et compter le nombre de caractères numériques de la suite.

Exercice 3 (4 pts)

Ecrire une *procédure* qui reçoit le rang dans l'alphabet d'un caractère alphabétique et fournit le caractère majuscule et le caractère minuscule correspondant.

Exercice 4 (2 pts)

Ecrire une *fonction* qui détermine le minimum de 3 nombres donnés.

Exercice 5 (2pts)

Ecrire un *algorithme* qui calcule le produit de deux matrices carrées de dimension n.

Exercice 6 (5pts)

Ecrire un *algorithme* qui simule le problème suivant:

Deux joueurs lancent un dé, Le joueur qui a le plus grand résultat marque un point. On arrête le jeu lorsque l'un des joueurs atteint 11.

Correction Examen final 2

Partie I : Questions de cours (2pts)

- 1) Quelle est la différence entre la boucle *tant que* et la boucle *répéter* ?
Avec le schéma « répéter » le bloc d'instructions qui forme le corps de la boucle est exécuté au moins une fois car la première exécution n'est soumise à aucune condition.
Avec le schéma « tant que » le bloc d'instructions qui forme le corps de la boucle peut ne jamais être exécuté (cas où la condition est initialement fausse)
- 2) A quoi servent les fonctions *round*, *ord* et *chr* ?
Round() : fonction qui fournit l'entier le plus proche
Round(6.7)=7
Ord (c) : fonction ordinal elle renvoie un entier positif ou nul correspondant au rang du caractère c, cet entier représente le code du caractère c.
Chr (i) : fonction caractère c'est la fonction inverse de ord pour un argument entier positif ou nul i, elle engendre le caractère de rang i.
- 3) Définissez la notion de *procédure* en algorithmique et donnez un exemple ?
Une procédure est la résolution d'un sous-problème
Elle est définie par :
L'entête de la procédure dont la syntaxe est la suivante
Procédure identificateur de procédure (liste des paramètres formels)
Environnement
Algorithme
Exemple
- 4) Que représente une *matrice* en algorithmique ?
Une matrice est un tableau à deux dimensions L et C avec L constitue le nombre de lignes de la matrice et C le nombre de colonnes de la matrice.
On dit qu'une matrice est une matrice carrée d'ordre n si $L=C=n$

Partie Exercices

Exercice 1

Partie 1 : Cet algorithme remplit une matrice ou tableau (2,2) comme suit

$$X = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

i=1 X(1,1)=1

j=1 val =val+1=2

i=1 X(1,2)=2

j=2 val =val+1=3

i=2 X(2,1)=3

j=1 val =val+1=4

i=2 X(2,2)=4

j=2 val =val+1=5

Partie 2 :

Cet algorithme remplit une matrice ou tableau (3,2) comme suit

$$T = \begin{bmatrix} 6 & 10 \\ 7 & 11 \\ 8 & 12 \end{bmatrix}$$

Exercice 3 (3 pts)

Algorithme qui permet de lire une suite de caractères

Le compteur n des caractères numériques est initialisé à 0

Algorithme caractères numériques

Variable c caractère, n entier

Début

$n \leftarrow 0$;

lire(c)

tant que($c \neq ' '$) faire

si ($c \geq '0'$ et $c \leq '9'$) **alors**

$n \leftarrow n+1$

finsi

 lire(c)

fin faire

écrire (n)

Fin

Exercice 4 (4 pts)

Procédure qui reçoit le rang dans l'alphabet d'un caractère alphabétique et fournit le caractère majuscule et le caractère minuscule correspondant.

Procédure chr_alphabétique(\downarrow rang :entier ; \uparrow maj : 'A'..'Z' ; \uparrow min : 'a'..'z')

Variable r entier

Début

r ← rang - 1 ;

maj ← chr(ord('A') + r)

min ← chr(ord('a') + r) ;

Fin

Exercice 5 (2 pts)

fonction qui détermine le minimum de 3 nombres donnés

Fonction minimum(\downarrow x,y :réel) :réel

Début

Si x < y Alors

minimum ← x

sinon

minimum ← y

finsi

Fin

Algorithme minimum

Variabes a,b,c réel

minimum fonction

Début

Ecrire('donnez les trois nombres :')

Lire(a,b,c)

Ecrire (minimum (minimum(a,b),c)

Fin

Exercice 6 (2pts)

Produit de deux matrices carrées de dimension n.

```
Produit_matriciel (a: Matrice carrée , b: Matrice carrée, n:
entier): Matrice carrée
VAR c: Matrice carrée n*n
    i: entier
Debut
    Pour i <- 1 a n Faire
        Pour j de 1 a n Faire
            c[i][j] <- 0
            Pour k de 1 a n Faire
                c[i][j] <- c[i][j] + a[i][k] * b[k][j]
            Fpour
        Fpour
    Fpour
    retourner c
```

Fin

Exercice 7 (5pts)

Algorithme jeu dedé

Variables nom1,nom2 chaines de caractères

Variables dé1,dé2,S1,S2 entier

Début

Ecrire('donnez noms des deux joueurs :')

Lire (nom1,nom2)

S1←0 ;

S2←0 ;

Répéter

 Ecrire ('donnez le dé de ',nom1) ;

 Lire(dé1) ;

 Ecrire ('donnez le dé de ',nom2) ;

 Lire(dé2) ;

 Si (dé1>dé2) Alors

 S1←S1+1 ;

 Sinon

Si (dé2>dé1) Alors

S2←S2+1

Fin si

 Fin si

Jusqu'à (S1=11 ou S2=11)

Si (S1=11) Alors

 Ecrire ('le gagnant est :',nom1)

sinon

 Ecrire ('le gagnant est :',nom2)

Finsi

Fin