

Exercice 1 (3,25 points)

1) Le tableau de déclaration du type temps

<i>Type</i>
<i>Temps = Enregistrement</i>
<i>Heures : 0..23</i>
<i>Minutes : 0..59</i>
<i>Secondes : 0..59</i>
<i>Fin Temps</i>

2) Les traitements demandés

Traitements	Instructions algorithmiques
Saisir la variable T1, en respectant les contraintes nécessaires	<i>Répéter</i> <i>Lire(T1.Heures)</i> <i>Jusqu'à T1.Heures dans [0..23]</i> <i>Répéter</i> <i>Lire(T1.Minutes)</i> <i>Jusqu'à T1.Minutes dans [0..59]</i> <i>Répéter</i> <i>Lire(T1.Secondes)</i> <i>Jusqu'à T1.Secondes dans [0..59]</i>
Affecter le contenu de la variable T1 à la variable T2	<i>T2 ← T1</i> <i>Ou bien</i> <i>T2.Heures ← T1.Heures</i> <i>T2.minutes ← T1.Minutes</i> <i>T2.Secondes ← T1.Secondes</i>
Afficher le contenu de la variable T2	<i>Ecrire ("Heures : ", T2.Heures)</i> <i>Ecrire ("Minutes : ", T2.Minutes)</i> <i>Ecrire ("Secondes : ", T2.Secondes)</i>

Exercice 2 (3 points)

0) ***DEF FN CalcSom(x,a : Réel) : Réel***

1) ***S ← x***

Terme ← x

Répéter

Terme ← - Terme*carré(x)/carré(a)

S ← S + Terme

Jusqu'à Abs(terme) < 0.0001

2) ***CalcSom ← S***

3) ***Fin CalcSom***

Exercice N°3 (3 points)

o) DEF PROC Tri (Var T : tab ; n : entier)

1) Pour i de 1 à ***n Div 2*** Faire

pmin ← i

pmax ← i

Pour j de i+1 à ***n-i+1*** Faire

Si (T[j] < T[pmin]) Alors pmin ← j

Sinon ***Si (T[j] > T[pmax]) Alors pmax ← j***

Fin si

Fin si

Fin pour

aux ← T[i]

T[i] ← T[pmin]

T[pmin] ← aux

Si (pmax=i) Alors pmax ← pmin

Fin Si

aux ← T[n-i+1]

T[n-i+1] ← T[pmax]

T[pmax] ← aux

Fin pour

2) Fin Tri

Exercice 4 (4,5 points)

DEF PROC Symétrie (Var M : Mat ; N :Entier)

Résultat = M

M = [M[0,1] ← 1]

Pour L de 1 à N-1 Faire

M[L,1] ← 1, M[L,L+1] ← 1

M[-L,1] ← M[L,1],

M[-L,L+1] ← M[L,L+1]

Pour C de 2 à L Faire

M[L,C] ← M[L-1,C] + M[L-1,C-1],

M[-L,C] ← M[L,C]

Fin Pour

Fin Pour

T.D.O.L

Objet	Type/nature	Rôle
L	Entier	Compteur
C	Entier	Compteur

Traitement
Entête : paramètres et passage
Initialisations
Parcours
Remplissage de la partie inf
Remplissage de la partie sup
T.D.O.L

Exercice 5 (6,25 points)

Analyse du programme principal "décryptage"

Résultat = Écrire (Msg)

Msg ← Fn Message(F)

F = Associer(F, "MsgCrypte.txt")

T.D.O.G

Objet	Type/nature	Rôle
Msg	Chaîne	
F	Texte	
Message	Fonction	

Analyse de la fonction Message

DEF FN Message(Var F : Texte) : Chaîne

Résultat = Message

Message ← sous_chaine(M, 1, Long(M)-1)

M = [M ← "", Ouvrir(F)]

Tant que Non(Fin_Fichier(F)) Faire

Lire_nl(F, Ligne)

M ← M + FN DetMot(Ligne) + " "

Fin Tant que

T.D.O.L

Objet	Type/nature	Rôle
M	Chaîne	
Ligne	Chaîne	
DetMot	Fonction	

Analyse de la fonction DetMot

DEF FN DetMot(Ch :Chaine) :Chaine

Résultat = DetMot

DetMot←P

P= [P← "", K←1]

Tant que K<Long(Ch) Faire

X←sous_chaine(Ch,K,8)

P←P+Chr(Fn Décimal(X))

K←K+8

Fin Tant que

T.D.O.L

Objet	Type/nature	Rôle
P	Chaine	
K	Entier	
X	Chaine	
Decimale	Fonction	

Analyse de la fonction Décimal

DEF FN Décimal(X :Chaine) : Octet

Résultat = Décimal

Décimal ← Dec

Dec =[Dec←0]

Pour J de 1 à 8 Faire

bit←Ord(X[J])-Ord("0")

Dec←Dec*2+bit

Fin Pour

T.D.O.L

Objet	Type/nature	Rôle
Dec	Octet	
J	Octet	
Bit	Octet	