

Exercice 1

PROCEDURE note-etudiant()

DPR

note : entier

FAIRE

DTQ

LIRE(note)

FTQ

TANT QUE (note < 0 OU note > 20)

SI (note < 10) ALORS ECRIRE("Insuffisant")

SINON SI (note < 12) ALORS ECRIRE("Passable")

SINON SI (note < 14) ALORS ECRIRE("AB")

SINON SI (note < 16) ALORS ECRIRE("B")

SINON ECRIRE("TB")

FPR

On choisit de faire une procédure car nous n'avons pas besoin de faire sortir un paramètre. Tout est géré à l'intérieur de la procédure.

Exercice 2

PROCEDURE table()

DPR

i, chiffre, res : entier

res ← 0

LIRE(chiffre)

POUR i DE 1 A 10 [PARPASDE 1]

DP

res ← chiffre * i

ECRIRE(chiffre, "X", i, "=", res)

FP

FPR

Exercice 3

PROCEDURE tab-tri(N: entier, tab: tableau[N] de entier)

DPR

i, j, tmp: entier

POUR i DE 1 A N [PARPASDE 1]

DP

j ← i

TANT QUE (j > 0 ET tab[j-1] > tab[j]) FAIRE

DTQ

tmp ← tab[j]

tab[j] ← tab[j-1]

tab[j-1] ← tmp

j ← j-1

FTQ

FP

FPR

Exercice 4

PROCEDURE remplissage - tab (T: tableau [12][8] de entier)

DPR

lig, col, val : entier

POUR lig DE 0 A 11 [PARPASDE 1]

DP

POUR col DE 0 A 7 [PARPASDE 1]

DP

Ecrire ("valeur ligne", lig, "colonne", col, ":")

LIRE (val)

T[lig][col] ← val

FP

FP

FPR

PROCEDURE plus-grande-val (T: tableau [12][8] de entier,
x: entier, y: entier)

DPR

lig, col : entier

x ← 0

y ← 0

POUR lig DE 0 A 11 [PARPASDE 1]

DP

POUR col DE 0 A 7 [PARPASDE 1]

DP

SI (T[x][y] < T[lig][col]) ALORS

DSI

x ← lig

y ← col

FSI

FP

FP

FPR

PROGRAMME PRINCIPAL

DPP

T: tableau [12][8] de entier

lig, col : entier

remplissage - tab (T)

plus-grande-val (T, lig, col)

Ecrire ("la plus grande valeur est aux coordonnées [", lig, ";", col, "]")

FPP

Exercice 5

5.1

Type

employe = enregistrement

nom : tableau [20] de caractere

prenom : tableau [20] de caractere

salaire : entier

anciennete : entier

fn.enregistrement

5.2

NB_EMPLOYE \leftarrow 5 : entier

base - employe : tableau [NB_EMPLOYE] de employe

5.3

PROCEDURE saisie - employe (inEmploye : employe)

DPR

ECRIRE ("nom de l'employe : ")

LIRE (inEmploye . nom [0])

ECRIRE ("prenom de l'employe : ")

LIRE (inEmploye . prenom [0])

ECRIRE ("salaire de l'employe : ")

LIRE (inEmploye . salaire)

ECRIRE ("anciennete de l'employe : ")

LIRE (inEmploye . anciennete)

FPR

5.4

PROCEDURE maj - salaire (size : entier, base_employes : tableau [size] de employe)

DPR

i : entier

POUR i DE 0 A size [PAR PAS DE 1]

DP

SI (base - employes [i] . anciennete > 4) ALORS

DSI

base - employes [i] . salaire \leftarrow (base - employes [i] . salaire * 1,02) + 200

FSI

FP

FPR

5.5

PROGRAMME PRINCIPAL

DPP

i, j, tmp : entier

tab : tableau [NB-EMPLOYE][2] de entier

POUR i DE 0 A NB-EMPLOYE [PAR PAS DE 1]

DP

ECRIRE ("Employe n°", i)

saisie-employe (base-employes[i])

tab[i][0] ← base-employes[i].anciennete

tab[i][1] ← i

FP

maj-salaire (NB-EMPLOYE, base-employes)

POUR i DE 1 A NB-EMPLOYE [PAR PAS DE 1]

DP

j ← i

TANT QUE (j > 0 ET tab[j-1][0] > tab[j][0])

DTQ

tmp ← tab[j][0]

tab[j][0] ← tab[j-1][0]

tab[j-1][0] ← tmp

tmp ← tab[j][1]

tab[j][1] ← tab[j-1][1]

tab[j-1][1] ← tmp

j ← j-1

FTQ

FP

FPP

// 0 par l'ancienneté
// 1 par le rang

// on change l'ordre de
l'ancienneté

// on change l'ordre du
rang

5.6

Type

date-embauche = enregistrement

jour : entier

mois : entier

annee : entier

fin-enregistrement

Type

employe = enregistrement

nom : tableau [20] de caractere

prenom : tableau [20] de caractere

salaire : entier

anciennete : date-embauche

fin-enregistrement

L'instruction qui permet de saisir l'année de l'embauche

unEmploye.anciennete.annee