

Bac 2015 Corrigé : Informatique

Sections : Math, Techniques et Sciences Expérimentales

Exercice 1 (1.25 x 3 = 3.75 points)

Afin de réaliser les tâches décrites dans la première colonne du tableau suivant, un élève fournit les propositions suivantes. Apporter les corrections nécessaires pour que ces propositions soient les plus adéquates relativement au choix de la structure itérative.

Tâche	Proposition	Correction
Saisir un entier n positif.	n= [n= donnée ("Saisir un entier positif :")] Tant que (n<0) faire n= donnée ("Saisir un entier positif :") Fin tant que	n=[] Répéter n= donnée ("Saisir un entier positif :") Jusqu'à (n ≥ 0)
Chercher la valeur maximale dans un tableau T de taille n .	Max = [i ← 1, Max ← t[1]] Répéter [] Si (T [i]> Max) alors Max ←T[i] Fin Si i←i+1 Jusqu'à (i > n)	Max = [Max← T[1]] Pour i de 2 à n faire Si (T[i]> Max) alors Max ←T[i] Fin Si Fin pour
Vérifier l'existence d'un caractère C dans un tableau T de n caractères.	Trouve= [Trouve ←faux] Pour i de 1 à n faire [] Si (T[i] = C) Alors Trouve ←Vrai Fin si Fin pour	Trouve= [i ← 0] Répéter i←i+1 Trouve ← T[i] = C Jusqu'à (Trouve) ou (i= n)

Exercice 2 (0.75 x 7 = 5.25 points)

Pour chacune des questions suivantes, cochez la ou les bonnes réponses.

1- Par quel appel peut-on remplacer la séquence 1 de cet algorithme ?

- Proc Saisir (N)
 Procédure Saisir (Var N : entier)
 Proc Saisir (U0)
 $U0 \leftarrow$ Proc saisir (N)

2- Quelles sont les entêtes qui correspondent à la procédure **Afficher**?

- DEF Proc Afficher (Var T : tab)
 DEF Proc Afficher (T : tab ; N : entier)
 DEF Proc Afficher (i : entier ; T : tab)
 DEF Proc Afficher (T[i] : entier)

3- L'entête suivante de la fonction Max est erronée : DEF FN **Max** (X : entier)

Quel est l'origine de l'erreur ?

- Le mode de passage des paramètres est erroné.
 Le nom du paramètre effectif est différent du nom du paramètre formel.
 Le type du résultat manque.
 Le type du paramètre effectif est incompatible avec celui du paramètre formel.

4- Si on veut remplacer la séquence 2 par l'appel d'un module :

a. Quelle sera sa nature ?

- Procédure Fonction

b. Quels seront les paramètres à utiliser ?

- T, i et U0
 T[i] et U0
 T et U0
 T et i

5- Quel sera le tableau de déclaration des objets de l'algorithme Suite ?

Objet	Type
T	Tab
U0	Entier

Objet	Type
T	Tab
I, U0	Entier
Max, Min	Fonction
Afficher	Procédure

6- Pour une valeur donnée d'U0 égale à **5360**. Quel sera le résultat de l'affichage de l'algorithme **Suite** ?

T

5843	5085	7992	7173	6354	3087	8352	6147	6174
------	------	------	------	------	------	------	------	------

T

5843	5085	2970	6930	5940	4950	4950
------	------	------	------	------	------	------

Problème (11 points)**TDNT****a) ANALYSE DU PROGRAMME PRINCIPAL****Résultat** = PROC **afficher** (T, n)T = PROC **trier** (T, nb)T, nb = PROC **ranger** (T, n, nb)T, n = PROC **remplir** (T, n)

Type
Vect = tableau de 44 entiers

TDOG

Objet	T/N
T	Vect
n, nb	Entier
Remplir	Procédure
Ranger	Procédure
Trier	Procédure
Afficher	Procédure

b) ANALYSE DE LA PROCEDURE REMPLIR**DEF PROC remplir** (var t : vect ; var n : entier)**Résultat** = t, n

t = []

Pour i de 1 à n faire

Répéter

t[i] = donnée ("saisir un entier :")

Jusqu'à (t [i] > 1)

Fin pour

n = []

Répéter

n = donnée ("saisir un entier :")

Jusqu'à (n dans [10..44])

Objet	Type
i	Entier

c) ANALYSE DE LA PROCEDURE RANGER**DEF PROC Ranger** (Var t : vect ; n : entier ; Var nb: entier)**Résultat** = t, nb

(t, nb) = [nb ← 0]

Pour i de 1 à n faire

Si (FN Premier (t[i]) et (FN Premier ((t[i]-1) div 2) alors

nb ← nb+1

aux ← t[i]

t[i] ← t[nb]

t[nb] ← aux

Fin si

Fin pour

Fin Ranger

Objet	Type
Premier	Fonction
PremSur	Fonction
i, j	Entier
aux	Entier

d) ANALYSE DE LA FONCTION**DEF FN Premier (r : entier) : booléen****Résultat** = premier \leftarrow (r mod d=0)d = [d \leftarrow 1]

Répéter

d \leftarrow d+1

Jusqu'à (r mod d =0) ou (d > racinecarré(r))

Fin premier

Objet	Type
d	Entier

e) ANALYSE DE LA PROCEDURE TRIER**Def Proc TRIER (VAR T : VECT ; N : ENTIER)****Résultat** = T

T = [] Répéter

Echange \leftarrow faux

Pour i de 1 à n-1 faire

Si (T[i] > T[i+1]) alors

Aux \leftarrow T[i]T[i] \leftarrow T[i+1]T[i+1] \leftarrow AuxEchange \leftarrow vrai

Fin si

Fin pour

n \leftarrow n-1

Jusqu'à (n=1) ou (Echange = faux)

Objet	Type
Aux	Entier
i	Entier
Echange	booléen