

Nom : OUARET Prénom : Ahmed Groupe : GP1

Exercice :

Soit l'algorithme suivant :

Algorithme	Programme C
Algorithme Test ; Variables T : Tableau [1.. 5] d'entiers ; nb, i : entiers ; Début Pour i ← 1 à 5 faire Lire (T[i]) ; FinPour ; nb ← 0 ; Pour i ← 1 à 4 faire Si (T[i] = T[5]) alors nb ← nb + 1 ; FinSi ; FinPour ; Ecrire('nb = ', nb) ; Fin.	<pre> #include <stdio.h> 0,1 int main() { 0,1 int T[5]; 0,1 int nb, i; 0,1 for (i = 1; i <= 5; i++) 0,1 scanf("%d", &T[i]); 0,1 nb = 0; 0,1 for (i = 1; i <= 4; i++) 0,1 if (T[i] == T[5]) 0,1 nb = nb + 1; 0,1 printf("nb = %d", nb); 0,1 return 0; 0,1 } 0,1 </pre>

Questions :

- Traduire l'algorithme en programme en langage C.
- Dérouler l'algorithme pour : T = [3, 1, 3, 5, 3]
- Déduire ce que fait l'algorithme ?
- Ré-écrire le programme en remplaçant la boucle *for* par la boucle *do-while* dans la partie de traitement.

Réponses :

Ce que fait l'algorithme :

L'algorithme permet de calculer le nombre de composantes égales à la dernière composante du tableau (sans compter la dernière composante)

Déroulement :

Instruction	Variables			Affichage					
	T	i	nb						
pour $i \leftarrow 1$ à 5 faire lire (T[i]); fin pour	<table border="1"><tr><td>3</td><td>1</td><td>3</td><td>5</td><td>3</td></tr></table>	3	1	3	5	3			
3	1	3	5	3					
$nb \leftarrow 0$			0						
pour $i \leftarrow 1$ si (T[1] = T[5]) alors 3 = 3 <u>vrai</u> $nb \leftarrow nb + 1 = 0 + 1 = 1$		1	1						
pour $i \leftarrow 2$ si (T[2] = T[5]) alors 1 = 3 <u>faux</u>		2							
pour $i \leftarrow 3$ si (T[3] = T[5]) alors 3 = 3 <u>vrai</u> $nb \leftarrow nb + 1 = 1 + 1 = 2$		3	2						
pour $i \leftarrow 4$ si (T[4] = T[5]) alors 5 = 3 <u>faux</u>		4							
Ecrire ("nb = ", nb);				nb = 2					

0,25
0,25
0,25
0,25
0,25
0,25

Ré-écrire le programme en remplaçant la boucle *for* par la boucle *do-while* dans la partie de traitement.

Programme Pascal	
pour i ← 1 à 5 faire lire (T[i]); fin pour	$nb = 0 ; i = 1 ;$ Do { if (T[i] == T[5]) $nb = nb + 1 ;$ $i = i + 1 ;$ } while (i ≤ 4);

②

Bonne chance