

Algorithmique II

Examen final

Corrigé

Exercice 1 : (Sur 7 points)

1. Déclarations

Type Etudiant= Structure

Code_Apogee : Caractere[1..8]

Nom : Caractere[1..10]

Prénom : Caractere[1..10]

Notes : Réel[1..6]

Fin Structure

Var T : Etudiant[1..N]

2. Procédure Résultat_SMI_S3(T : Etudiant[1..N])

Var i, j : Entier

M : Réel

Valide : boolleen

Début

Ecrire("\nCode_apogee \t Nom \t Prénom \t Moyenne \t Résultat\n")

Pour i←1 à N Faire

M←0

Pour j←1 à 6 Faire

M←M+ T[i].Notes[j]

Fin Pour

M←M/6

Ecrire("\n",T[i].Code_apogee,"\t", T[i].Nom,"\t", T[i].Prénom,"\t", M,"\t")

Valide ← VRAI

Pour j←1 à 6 Faire

Si T[i].Notes[j]<5 Alors

Valide←FAUX

Finsi

Fin Pour

Si Valide=VRAI et M>=10 Alors

Ecrire("Semestre Validé \n")

Sinon

Ecrire("Semestre non validé \n")

FinSi

Fin Pour

Fin

Exercice 2 : (Sur 6 points)

```
Fonction : PrefSuff(T : Entier[1..n], l : entier) : boolleen
//Données : T[1..n] un tableau d'entiers, l ∈ [1, n-1]
//Résultat : Retourne Vrai si T[1..l] est en même temps préfixe et suffixe de T,
//retourne Faux sinon
Var    j : Entier
Début
    j ← 1
    tant que (j ≤ l et T[j] = T[n-l + j]) faire
        j ← j + 1
    Fin tant que
    Si (j=l+1) Alors
        Retourner Vrai
    Sinon
        Retourner Faux
    FinSi
Fin

Fonction PlusGrandPrefSuff(T : Entier[1..n]) : Entier
//Données : T[1..n] un tableau d'entiers
//Résultat : Retourne la longueur du plus grand préfixe et suffixe de T
Var    i, max : Entier
Début
    max ← 0
    Pour i ← 1 à n-1 Faire
        Si PrefSuff(T,i) Alors
            max ← i
        Fin si
    Fin Pour
    Retourner max
Fin
```

Ou encore :

```
Fonction PlusGrandPrefSuff(T : Entier[1..n] ) : Entier
//Données : T[1..n] un tableau d'entiers
//Résultat : Retourne la longueur du plus grand préfixe et suffixe de T
Var    i, j, max : Entier
Début
    max ← 0
    Pour i ← 1 à n-1 faire
        j ← 1
        Tant que j ≤ i et T[j] = T[n- i + j] faire
            j ← j + 1
        Fin Tant que
        Si j = i + 1 Alors
```

```

        max ← i
    Fin Si
Fin Pour
Retourner max
Fin

```

Exercice 3 : (Sur 7 points)

On considère les deux fonctions récursives F1 et F2 données par :

Fonction F1(n : Entier) : Entier Début Si (n = 0) Alors Retourner(2) Sinon Retourner(F1(n - 1) * F1(n - 1)) Fin Si Fin	Fonction F2(n : Entier) : Entier Var m : Entier Début Si (n = 0) Alors Retourner(2) Sinon m ← F1(n - 1) Retourner(m * m) Fin Si Fin
--	--

1. On montre par récurrence sur n que les deux fonctions F1(n) et F2(n) retournent 2^{2^n}
2. Le nombre de multiplications $m_1(n)$ effectuées par F1 est solution de l'équation de récurrence :

$$m_1(n) = 0 \text{ si } n = 0 \text{ et}$$

$$m_1(n) = 1 + 2 * m_1(n - 1) \text{ si } n > 0$$

On montre facilement par récurrence que :

$$m_1(n) = \sum_{i=0}^{n-1} 2^i = 2^n - 1$$

On en déduit que F1(n) est en $\Theta(2^n)$ et possède donc une complexité exponentielle.

Le nombre de multiplications $m_2(n)$ effectuées par F2 est solution de l'équation de récurrence :

$$m_2(n) = 0 \text{ si } n = 0 \text{ et}$$

$$m_2(n) = 1 + m_2(n - 1) \text{ si } n > 0$$

On montre facilement par récurrence que :

$$m_2(n) = \sum_{i=0}^{n-1} 1 = n$$

On en déduit que F2(n) est en $\Theta(n)$ et possède donc une complexité linéaire