**CORRIGÉ INFORMATIQUE SECTIONS SCIENTIFIQUES**

**SESSION PRINCIPALE BACCALAUREAT 2017**

**Exercice N°1 : (2.5 points)**

Réécrire dans la colonne « **Correction »**,les phrases données dans la colonne « **Proposition** » en apportant les modifications nécessaires afin qu’elles soient correctes :

|  |  |
| --- | --- |
| **Proposition** | **Correction** |
| La recherche dichotomique est une méthode de recherche appliquée sur un tableau quelconque. | * La recherche dichotomique est une méthode de recherche appliquée sur un tableau **trié**. * La recherche **séquentielle** est une méthode de recherche appliquée sur un tableau quelconque. |
| Un tableau est une structure de données permettant de regrouper plusieurs éléments de types différents. | Un tableau est une structure de données permettant de regrouper plusieurs éléments de **même** type. |
| Les paramètres formels sont utilisés lors de l’appel d’un sous-programme. | * Les paramètres formels sont utilisés lors de **la définition** d’un sous-programme **(au niveau de l’entête)**. * Les paramètres **effectifs** sont utilisés lors de l’appel d’un sous-programme. |
| L’évaluation d’une expression formée par des opérateurs ayant la même priorité se fait de gauche à droite ou de droite à gauche. | L’évaluation d’une expression formée par des opérateurs ayant la même priorité se fait de **gauche à droite**. |
| Le type scalaire énuméré définit un ensemble ordonné et fini de valeurs appartenant à un type prédéfini. | * Le type scalaire énuméré définit un ensemble ordonné et fini de valeurs **n’appartenant pas à un type prédéfini** **dites identificateurs**. * Le **type intervalle** définit un ensemble ordonné et fini de valeurs appartenant à un type prédéfini. |

**Exercice 2 : (5.5 points)**

|  |  |  |
| --- | --- | --- |
| **T.D.O** | | |
| **Objet** | **Type/nature** | **Rôle** |
| ch1 | Chaîne | La première chaîne à saisir |
| ch2 | Chaîne | La deuxième chaîne à saisir |
| p | Entier | Position |
| ch | Chaîne | Chaîne intermédiaire |
| i | Entier | Compteur |

1. Séquence n°2 : ***ch🡨 Sous-chaine (ch1,1,p-1)***

Séquence n°4 : ***ch🡨 Concat (ch, Sous-chaine(ch1,p, Long(ch1)-p+1))***

1. Le programme affiche : ***BAC 2016/2017***
2. Cet algorithme permet **d’insérer** une chaine **ch2** dans une chaine **ch1** à une position **p**.
3. Insère (ch2, ch1, p)

**Problème :** **(12 points)**

1. **Analyse du programme principal**

|  |
| --- |
| ***TDNT*** |
| ***Type*** |
| *Tab1= tableau de 5 chaînes* |
| *Tab2= tableau de 5 entiers* |

*Nom : Hitparade*

*Résultat= PROC Afficher (T1, T2)*

*(T1,T2)= [PROC Chanson (T1)*

*PROC Score (T2,N)]*

*PROC Trier (T1,T2)*

*N= PROC Saisir(N)*

*Fin Hitparade*

|  |  |  |
| --- | --- | --- |
| ***TDOG*** | | |
| ***Nom*** | ***Nature/Type*** | ***Rôle*** |
| *T1* | *Tab1* | *Tableau des chansons* |
| *T2* | *Tab2* | *Tableau des scores* |
| *N* | *Entier* | *Nombre de participants* |
| *Saisir* | *Procédure* | *Saisir le nombre des participants* |
| *Chanson* | *Procédure* | *Remplir le tableau des chansons* |
| *Score* | *Procédure* | *Comptabiliser les scores* |
| *Trier* | *Procédure* | *Trier les scores et les chansons* |
| *Afficher* | *Procédure* | *Afficher le résultat* |

1. **Les algorithmes des modules envisagés :**
2. **Module *Chanson :***
   1. ***DEF PROC Chanson (Var T1 : Tab1)***
   2. *Pour i de 1 à 5 faire*

*Répéter*

*Ecrire (" Saisir le titre de la chanson n°", i, " :")*

*Lire (T1[i])*

*Jusqu’à (FN Verif (T1[i]))*

*Fin pour*

* 1. ***Fin Chanson***

|  |  |  |
| --- | --- | --- |
| ***TDOL*** | | |
| ***Nom*** | ***Nature/Type*** | ***Rôle*** |
| *i* | *Entier* | *Compteur* |
| *Verif* | *Fonction* | *Vérifier si une chaîne est composée uniquement de lettres et des espaces* |

1. **Module *Verif :***
   1. *DEF FN**Verif* ***(****ch : chaine) : booléen*
   2. *j 🡨 0*

*Répéter*

*j🡨 j + 1*

*Jusqu’à (NON (Majus (ch[j]) dans ["A".."Z", " "])) OU ( j=Long(ch))*

* 1. ***V****erif🡨 (Majus (ch[j]) dans ["A".."Z", " "]*
  2. *Fin Verif*

|  |  |  |
| --- | --- | --- |
| ***TDOL*** | | |
| ***Nom*** | ***Type*** | ***Rôle*** |
| *j* | *Entier* | *Compteur* |

1. **Module *Saisir :***
2. *DEF PROC Saisir (Var N : entier)*
3. *Répéter*

*Ecrire (" Saisir le nombre de participants : ")*

*Lire (N)*

*Jusqu’à (N dans [5..100])*

1. *Fin Saisir*
2. **Module *Score :***
3. *DEF PROC Score (Var T2 : Tab2 ; N : entier )*
4. *Pour i de 1 à 5 faire*

*T2[i] 🡨 0*

*Fin pour*

1. *Pour i de 1 à N faire*

*Répéter*

*Ecrire (" Saisir le numéro de la chanson :")*

|  |  |  |
| --- | --- | --- |
| ***TDOL*** | | |
| ***Nom*** | ***Nature/Type*** | ***Rôle*** |
| *i* | *Entier* | *Compteur* |
| *num* | *Entier* | *Numéro de la chanson choisie* |

*Lire (num)*

*Jusqu’à (num dans [1..5])*

*T2[num] 🡨 T2[num] + 3*

*Fin pour*

1. *Fin Score*
2. **Module *Trier :***
3. *DEF PROC Trier (Var T2 : Tab2 ; Var T1 : Tab1 )*

|  |  |  |
| --- | --- | --- |
| ***TDOL*** | | |
| ***Nom*** | ***Type*** | ***Rôle*** |
| *i* | *Entier* | *Compteur* |
| *Pmax* | *Entier* | *Position maximum* |
| *Aux1* | *Entier* | *Variable intermédiaire* |
| *Aux2* | *Chaine* | *Variable intermédiaire* |
| *Posmax* | *Fonction* | *Renvoie la position du maximum* |

1. *Pour i de 1 à 4 faire*

*Pmax🡨FN Posmax (T2, i)*

*Si ( T2[Pmax]<>T2[i])*

*Alors*

*Aux1🡨 T2[Pmax]*

*T2[Pmax] 🡨 T2[i]*

*T2[i] 🡨 Aux1*

*Aux2🡨 T1[Pmax]*

*T1[Pmax] 🡨 T1[i]*

*T1[i] 🡨 Aux2*

*Fin Si*

*Fin pour*

1. *Fin Trier*
2. **Module *Posmax***
3. *DEF FN Posmax (T2 : Tab2 ; i : entier)*
4. *Pm🡨i*

|  |  |  |
| --- | --- | --- |
| ***TDOL*** | | |
| ***Nom*** | ***Nature/Type*** | ***Rôle*** |
| *j* | *Entier* | *Compteur* |
| *Pm* | *Entier* | *Position maximum* |

1. *Pour j de i+1 à 5 faire*

*Si (T2[Pm] < T2[j])*

*Alors Pm🡨j*

*Fin Si*

*Fin pour*

1. *Posmax 🡨 Pm*
2. *Fin Posmax*
3. **Module *Afficher :***
4. *DEF PROC Afficher (T1 : Tab1;T2 : Tab2 )*
5. *Ecrire ("le classement est :")*

*Rang🡨1*

|  |  |  |
| --- | --- | --- |
| ***TDOL*** | | |
| ***Nom*** | ***Nature/Type*** | ***Rôle*** |
| *i* | *Entier* | *Compteur* |
| *Rang* | *Entier* | *Rang de la chanson* |

*Ecrire ("Rang ",Rang, " :", T1[1])*

*Pour i de 2 à 5 faire*

*Si (T2[i-1] =T2[i] )*

*Alors Ecrire (" ,  ",T1[i])*

*Sinon*

*Rang 🡨Rang+1*

*Ecrire ("Rang ",Rang, " :",T1[i])*

*Fin Si*

*Fin pour*

1. *Fin Afficher*