**DEVOIR D’ALGORITHMIQUE**

**Exercice 1: (/10)**

 Ecrivez l’algorithme qui permet à un utilisateur d’entrer des valeurs **comprises en 1 et 99** dans un tableau d’entiers. Pour arrêter, la saisie, il suffira d’entrer la valeur 0. Le programme affichera alors le nombre de valeur saisies, le plus grand nombre, le plus petit, et la moyenne des valeurs saisies. (cf exemple d’affichage au tableau). Le nombre de valeurs saisies n’excèdera jamais 40.

**Exercice 2 (/5)**

Ecrivez l’algorithme qui permettra d’obtenir ce résultat à l’écran:

Nom de la personne 1: **BRASSENS**

Métier de la personne 1 : **CHANTEUR**

 Nom de la personne 2: **BACH**

Métier de la personne 2 : **MUSICIEN**

Nom de la personne 3: **MATISSE**

Métier de la personne 3 : **PEINTRE**

Nom de la personne 4: **ANDRE**

Métier de la personne 4 : **TROMPETTISTE**

Nom de la personne 5: **PEREC**

Métier de la personne 5 : **ATHLETE**

Nom de la personne 6: **SEGUELA**

Métier de la personne 6 : **PUBLICISTE**

Le métier de BRASSENS est CHANTEUR c’est sûr !

Le métier de BACH est MUSICIEN c’est sûr !

Le métier de MATISSE est PEINTRE c’est sûr !

Le métier de ANDRE est TROMPETTISTE c’est sûr !

Le métier de PEREC est ATHLETE c’est sûr !

Le métier de SEGUELA est PUBLICISTE c’est sûr !

***Nb***: Tous les termes en gras sont saisis par l’utilisateur.

**Exercice 3: (/5)**

Donnez le résultat de cet algorithme à l’écran:

**Action** CASSTET

**Var** I, J, A : Entier

**Début**

 A ⇐ 0

 **Pour** I **de** 1 **à** 3 **Faire**

 Afficher ‘’ P1: ‘’, A

 **Pour** J **de** 1 **à** 2 **Faire**

 A ⇐ (A + J) \* I

 Afficher ‘’P2: ‘’, A

 **Fpour**

 A ⇐ A + (I \* J)

 Afficher ‘’P3: ’’, A

 **Fpour**

**Fin**

***Remarque*:** Chaque ordre *Afficher* écrit sur la ligne suivante

**Correction du devoir d’algorithmique No 1**

**Exercice 1:**

**Action** Saisie\_Tab

Var T : Tableau [1..50] d’entiers

 Mini, Maxi, Total, Nb : Entiers

**Début**

 Nb ⇐ 1; Mini ⇐ 1000; Maxi ⇐ 0

 **Répéter**

 Afficher ‘’Saisissez un entier positif ou 0 pour terminer’’

 Saisir T[Nb]

 **Si** T[Nb] > 0 **alors**

 **Début** (\* On traite le cas... \*)

 **Si** T[Nb] > Maxi **alors** (\* c’est le plus grand jusqu’à présent ! \*)

 Maxi ⇐ T[Nb] (\* on remplace maxi par T[Nb] \*)

 **Fsi**

 **Si** T[Nb] < Mini **alors** (\* C’est le plus petit jusqu’à présent ! \*)

 Mini ⇐ T[Nb] (\* On remplace Mini par T[Nb] \*)

 **Fsi**

 Total ⇐ Total + T[Nb]

 Nb ⇐ Nb + 1 (\* On passe au suivant \*)

 **Fin**

 **Fsi**

 **Jusqu’à** (T[Nb] = 0)

 Afficher ‘’Il y a ‘’, Nb, ‘’ notes’’

 Afficher ‘’La plus petite est: ‘’, Mini

 Afficher ‘’La plus forte est: ‘’, Maxi

 Afficher ‘’La moyenne est: ‘’, Total / (Nb - 1)

**Fin**

**Exercice 2:**

**Action** Essai

Var NOM, MET : Tableau [1..6] de chaînes

 I : Entier

**Début**

 (\* Remplissage des tableaux ... \*)

 **Pour** I **de** 1 **à** 6 **Faire**

 Afficher ‘’Nom de la personne ’’, I

 Saisir NOM[I]

 Afficher ‘’Métier de la personne ‘’,I

 Saisir MET[I]

 **Fpour**

 (\* Affichage des tableaux ... \*)

 **Pour** I **de** 1 **à** 6 **Faire**

 Afficher ‘’Le métier de ‘’,NOM[I],’’ est ‘’,MET[I],’’ c’est sûr!’’

 **Fpour**

**Fin**

**Exercice 3:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **I** | **J** | **A** |  | **Affichage** |
| 1 | 1 | 0 |  | **P1: 0** |
| 2 | 2 | 1 |  | **P2: 1** |
| 3 | 1 | 3 |  | **P2: 3** |
|  | 2 | 5 |  | **P3: 5** |
|  | 1 | 12 |  | **P1: 5** |
|  | 2 | 28 |  | **P2: 12** |
|  |  | 32 |  | **P2: 28** |
|  |  | 99 |  | **P3: 32**  |
|  |  | 303 |  | **P1: 32** |
|  |  | 309 |  | **P2: 99** |
|  |  |  |  | **P2: 303** |
|  |  |  |  | **P3: 309** |

**Traduction et exécution de l’exercice 3 avec DELPHI**

Le composant utilisé pour l’affichage est une zone de liste (ListBox). On ajoute un composant en utilisant la méthode **Liste.Items.Add(ch)** où ***Liste*** est le nom du composant et ***Ch*** la chaîne à ajouter.



**procedure** TForm1.LancerClick(Sender: TObject);

Var I, J, A : Integer;

 CH : String;

**begin**

 A := 0;

 **For** I := 1 **To** 3 **Do**

 **Begin**

 Ch := 'P1: ' + IntToStr(A);

 Liste.Items.Add(Ch);

 **For** J := 1 **To** 2 **Do**

 **Begin**

 A := (A + J) \* I;

 Ch := 'P2: ' + IntToStr(A);

 Liste.Items.Add(CH);

 **End**;

 A := A + (I \* J);

 Ch := 'P3: ' + IntToStr(A);

 Liste.Items.Add(Ch);

 **end**;

**end**;