

Rappel : Pour chaque question, on établira un algorithme sous forme de pseudo-code sur feuille avant d'implémenter cet algorithme en C#.

On utilisera le formalisme suivant pour écrire un algorithme :

Algorithme <nom de l'algorithme>
Principe : <idée principale/principe fondamental de l'algorithme>
Entrée : <variables d'entrée sous la forme nom : type>
Local : <variables locales sous la forme nom : type>
Sortie : <variable de sortie sous la forme nom : type ou void>
Début
 <...>
Fin

On utilisera le formalisme suivant pour définir une structure :

Soit structure <nom de la structure>
{
 champ <nom : type>
}

Exercice 1 : Date

On souhaite représenter le concept de date (jour, mois et année).

- 1) Définir une structure Date.
- 2) Écrire un algorithme permettant de créer une date à partir des données nécessaires.
- 3) Écrire un algorithme permettant d'afficher une date. Résultat attendu :

```
-> Date : 02/10/2024
```

- 4) Écrire un algorithme permettant de créer une date seulement si les informations données forment une date valide (on tiendra compte des années bissextiles). On pourra s'inspirer du TD1, exercice 6.
- 5) Écrire un algorithme permettant de comparer deux dates. Résultat attendu :

```
-> xDate1 04/10/2024 > xDate2 01/10/2024
```

Exercice 2 : État civil

On souhaite représenter l'état civil simplifié d'un individu. On considérera les informations suivantes : nom de naissance, nom de famille, prénom, date de naissance, lieu de naissance, prénom du père, prénom de la mère.

- 1) Définir la structure EtatCivil adéquate. On pensera à réemployer la structure Date de l'exercice 1.
- 2) Écrire un algorithme permettant de générer l'état civil d'un individu à partir de ses parents et des informations nécessaires.
- 3) Écrire un algorithme permettant d'afficher l'état civil d'un individu donné. Résultat attendu :

```
-> EtatCivil : Albert Dupont (nom de famille : Dupont) né(e) le 12/12/1975 à Strasbourg.
```

On stockera désormais les individus dans un tableau. Remplir le tableau avec les individus suivants.

EtatCivil	Nom de naissance	Nom de famille	Prénom	Date de naissance	Lieu de naissance	Père	Mère
individu1	Béranger	Béranger	Alfred	12/09/1939	Ostwald	/	/
individu2	Dupuy	Béranger	Valérie	21/06/1942	Haguenau	/	/
individu3	Béranger	Béranger	Charles	01/02/1964	Ostwald	Alfred	Valérie
individu4	Bonhomme	Béranger	Aline	08/09/1965	Illkirch	/	/
individu5	Béranger	Béranger	Mathieu	25/05/1992	Strasbourg	Charles	Aline
individu6	Béranger	Béranger	Pascale	30/11/1995	Strasbourg	Charles	Aline
individu7	Duchamps	Duchamps	Paul	03/04/1963	Mulhouse	/	/
individu8	Chevalier	Duchamps	Isabelle	11/11/1966	Altkirch	/	/
individu9	Duchamps	Duchamps	Marie	12/05/1994	Mulhouse	Paul	Isabelle
individu10	Béranger	Béranger	Louis	01/07/2020	Strasbourg	Mathieu	Marie

- 4) Écrire un algorithme qui affiche tous les états civils stockés dans le tableau. Résultat attendu :

```
-> EtatCivil : Alfred Béranger (nom de famille : Béranger) né(e) le 12/09/1939 à Ostwald.  
-> EtatCivil : Valérie Dupuy (nom de famille : Béranger) né(e) le 21/06/1942 à Haguenau.  
-> EtatCivil : Charles Béranger (nom de famille : Béranger) né(e) le 01/02/1964 à Ostwald.  
-> EtatCivil : Aline Bonhomme (nom de famille : Béranger) né(e) le 08/09/1965 à Illkirch.  
-> EtatCivil : Mathieu Béranger (nom de famille : Béranger) né(e) le 25/05/1992 à Strasbourg.  
-> EtatCivil : Pascale Béranger (nom de famille : Béranger) né(e) le 30/01/1995 à Strasbourg.  
-> EtatCivil : Paul Duchamps (nom de famille : Duchamps) né(e) le 03/04/1963 à Mulhouse.  
-> EtatCivil : Isabelle Chevalier (nom de famille : Duchamps) né(e) le 11/11/1966 à Altkirch.  
-> EtatCivil : Marie Duchamps (nom de famille : Duchamps) né(e) le 12/05/1994 à Mulhouse.  
-> EtatCivil : Louis Béranger (nom de famille : Béranger) né(e) le 01/07/2020 à Ostwald.
```

- 5) Écrire un algorithme permettant d'afficher les parents d'un individu donné. Résultat attendu :

```
-> Recherche des parents de Mathieu Béranger :  
-> -> Charles Béranger est le père de Mathieu Béranger.  
-> -> Aline Bonhomme est la mère de Mathieu Béranger.
```

- 6) Écrire un algorithme permettant d'afficher les enfants d'un individu donné. Résultat attendu :

```
-> Recherche des enfants de Aline Bonhomme :  
-> -> Mathieu Béranger est l'enfant de Aline Bonhomme (mère).  
-> -> Pascale Béranger est l'enfant de Aline Bonhomme (mère).
```

Exercice 3 : Étudiants

On souhaite modéliser une promotion d'étudiants. Un étudiant sera représenté par un nom, un prénom et un numéro de groupe. On souhaite également stocker les résultats de chaque UE dans un tableau (on fixe le nombre d'UE à 6) et on gardera à jour le nombre de notes renseignées ainsi que la moyenne sur ces notes.

- 1) Écrire un algorithme permettant de créer un étudiant. Les notes sont initialisées à 0.
- 2) Écrire un algorithme permettant d'afficher les détails d'un(e) étudiant(e). Résultat attendu :

```
-> Étudiant(e) : Einstein Albert Groupe 1
    notes : < 0.00 0.00 0.00 0.00 0.00 0.00 > moyenne 0.00 (0 notes renseignées).
```

- 3) Écrire un algorithme permettant de renseigner la note de la prochaine UE. Résultat attendu :

```
-> Étudiant(e) : Einstein Albert Groupe 1.
    notes : < 10.00 0.00 0.00 0.00 0.00 0.00 > moyenne 10.00 (1 notes renseignées).
-> Étudiant Einstein Albert Groupe 1
    notes < 10.00 15.00 7.50 13.25 18.00 9.75 > moyenne 12.25 (6 notes renseignées).
```

- 4) On souhaite représenter une promotion de 6 étudiants dans un tableau initialisé avec les données :

Nom	Prénom	Groupe	Notes UE
Turing	Alan	1	12 – 15.5 – 17.50 – 12 – 10 – 18
Lovelace	Ada	1	14 – 14 – 15 – 15 – 16 – 16
Boole	George	1	18 – 19 – 20 – 8 – 10.5 – 12
Babbage	Charles	2	15 – 15 – 10 – 5.5 – 18 – 19
von Neumann	John	2	7.5 – 14.75 – 17.50 – 18.25 – 11.25 – 9.75
Créhange	Marion	2	12 – 14 – 18 – 9.5 – 11.25 – 20

Écrire un algorithme affichant tous les étudiants de la promotion. Résultat attendu :

```
-> Étudiant(e) : Alan Turing Groupe 1
    notes : < 12.00 15.50 17.50 12.00 10.00 18.00 > moyenne 14.17 (6 notes renseignées).
-> Étudiant(e) : Ada Lovelace Groupe 1
    notes : < 14.00 14.00 15.00 15.00 16.00 16.00 > moyenne 15.00 (6 notes renseignées).
-> Étudiant(e) : George Boole Groupe 1
    notes : < 18.00 19.00 20.00 8.00 10.50 12.00 > moyenne 14.58 (6 notes renseignées).
-> Étudiant(e) : Charles Babbage Groupe 2
    notes : < 15.00 15.00 10.00 5.50 18.00 19.00 > moyenne 13.75 (6 notes renseignées).
-> Étudiant(e) : John von Neumann Groupe 2
    notes : < 7.50 14.75 17.50 18.25 11.25 9.75 > moyenne 13.17 (6 notes renseignées).
-> Étudiant(e) : Marion Créhange Groupe 2
    notes : < 12.00 14.00 18.00 9.50 11.25 20.00 > moyenne 14.13 (6 notes renseignées).
```

- 5) Écrire un algorithme calculant la moyenne générale de la promotion. Résultat attendu :

```
-> La moyenne générale de la promo est de 14.132/20.
```

- 6) Écrire un algorithme permettant de calculer et d'afficher la moyenne de chaque groupe. Résultat attendu :

```
-> La moyenne générale du groupe 1 est de 14.583/20.
-> La moyenne générale du groupe 2 est de 13.681/20.
```

- 7) Écrire un algorithme permettant d'afficher l'identité de l'étudiant(e) ayant obtenu la meilleure note pour chaque UE. Résultat attendu :

```
-> Major de l'UE 0 : George Boole (groupe 1) avec une moyenne de 18.000/20.
-> Major de l'UE 1 : George Boole (groupe 1) avec une moyenne de 19.000/20.
-> Major de l'UE 2 : George Boole (groupe 1) avec une moyenne de 20.000/20.
-> Major de l'UE 3 : John von Neumann (groupe 2) avec une moyenne de 18.250/20.
-> Major de l'UE 4 : Charles Babbage (groupe 2) avec une moyenne de 18.000/20.
-> Major de l'UE 5 : Marion Créhange (groupe 2) avec une moyenne de 20.000/20.
```