

Algorithmique & Programmation

Semestre 2 ST

Les types agrégés

Rappels du semestre 1

<u>Types agrégés</u>	<u>Structuration en sous-algorithmes</u>	<u>Tableaux de variables</u>	<u>Algorithmes de recherche et de tri</u>	<u>Précision, rapidité et complexité</u>
<u>Matrices de variables</u>	<u>Récursivité</u>	<u>Travaux dirigés</u>	<u>Travaux pratiques</u>	<u>Informations</u>
<u>Sujets de projet</u>	<u>Evaluation intermédiaire</u>	<u>Evaluation finale</u>	<u>Evaluation complémentaire</u>	<u>Archives</u>
<u>Cours</u>	<u>TD - Corrections</u>		<u>TP</u>	
<u>Fichier PDF</u>				

Tous les exercices sont à réaliser en langage algorithmique.

Exercice n°1

Pour une fédération sportive, on souhaite développer un programme de gestion des licences.

Concevoir un type agrégé "licence" de stockage des informations relatives à un licencié:

- nom,
- prénom,
- numéro de licence,
- club,
- catégorie d'âge (poussin, benjamin, minime, cadet, junior, senior, vétérans),
- sexe.

Exercice n°2

a) Concevoir un type agrégé "position3D" de stockage d'une position définie dans un espace à trois dimensions réelles.

b) Concevoir un algorithme de lecture au clavier d'une variable de type position3D, puis d'affichage à l'écran de cette variable.

Exercice n°3

a) Concevoir un type agrégé "temps" de stockage d'une heure représentée par un nombre d'heures, de minutes et de secondes.

b) Concevoir un algorithme réalisant les traitements suivants:

- Lecture au clavier de deux variables de type temps.
- Affichage de l'heure la plus tardive parmi les deux, puis affichage de l'autre heure.
- Affichage du nombre de secondes entre les deux.

Pour comparer deux temps, on pourra les convertir en secondes et effectuer la comparaison sur les nombres de secondes obtenus.

Exercice n°4

a) Concevoir un type agrégé "couleur" de stockage d'une couleur représentée par une valeur de rouge, une valeur de vert et une valeur de bleu. Ces trois valeurs sont de type entier. Usuellement en informatique, elles sont comprises entre 0 et 255 (8 bits).

Exemples: (255,255,255) -> blanc, (0,0,0) -> noir, (255,0,0) -> rouge, (255,255,0) -> jaune

b) Concevoir un algorithme réalisant les traitements suivants:

- Lecture au clavier d'une variable de type couleur.
- Lecture au clavier d'un pourcentage d'assombrissement.
- Calcul et affichage de cette couleur après assombrissement de la valeur lue.
- Lecture au clavier d'un pourcentage d'éclaircissement.
- Calcul et affichage de cette couleur après éclaircissement de la valeur lue.

Pour assombrir une composante de couleur de n %, on pourra la multiplier par $(1.0-n/100.0)$.

Pour éclaircir une composante de couleur c de n %, on pourra lui appliquer la formule $f(c) = 255.0 - ((255.0 - c) * (1.0 - n/100.0))$.

Exercice n°5

a) Concevoir un type agrégé "position2D" de stockage d'une position définie dans un espace à deux dimensions réelles.

b) Concevoir un type agrégé "triangle2D" de stockage d'un triangle de 3 position2D.

c) Concevoir un algorithme réalisant les traitements suivants:

- Lecture au clavier d'un triangle2D.
- Calcul et affichage de la surface de ce triangle2D.

Pour calculer la surface d'un triangle défini par les 3 sommets (x_1, y_1) , (x_2, y_2) et (x_3, y_3) , on pourra utiliser la formule suivante:

$$\text{surface} = \text{abs}((x_1+x_2)*(y_1-y_2)+(x_2+x_3)*(y_2-y_3)+(x_3+x_1)*(y_3-y_1))/2.0$$

Exercice n°6

a) Concevoir un type agrégé "parallelepipede" de stockage des caractéristiques d'un parallélépipède rectangle à faces perpendiculaires aux axes.

b) Concevoir un algorithme réalisant les traitements suivants:

- Lecture d'un parallépipede.
- Calcul et affichage du volume de ce parallépipede.

Le volume d'un parallépipede est égal au produit de la longueur de ses cotés.

Exercice n°7

a) Concevoir un type agrégé "date" de stockage d'une date représentée par un numéro de jour, un numéro de mois et un numéro d'année.

b) Concevoir un algorithme réalisant les traitements suivants:

- Lecture d'une date.
- Calcul de la date du lendemain.
- Affichage de cette date.

Etablir la date du lendemain à partir d'une date quelconque nécessite une analyse pour savoir si le jour est le dernier du mois au quel cas, il faudra passer au premier du mois suivant. Le passage au mois suivant peut lui-même entraîner le passage à l'année suivante.

Etablir le nombre de jours d'un mois est un traitement relativement complexe car ce nombre est égal à 31 pour les mois de janvier, mars, mai, juillet, aout, octobre et décembre, à 30 pour les mois d'avril, juin, septembre et novembre, à 28 pour le mois de février des années non bissextiles et à 29 pour le mois de février des années bissextiles.

Une année n'est pas bissextile si son numéro n'est pas divisible par 4. Si son numéro est divisible par 4, une année est bissextile. Cette règle admet comme exceptions les années dont le numéro est divisible par 100 (qui ne sont donc pas bissextiles). Cette règle admet elle-même comme exceptions les années dont le numéro est divisible par 400 (qui sont donc bissextiles).

Exemples: 1983 n'est pas bissextile, 1980 est bissextile, 2100 n'est pas bissextile, 2000 est bissextile.

Exercice n°8

- a) Concevoir un type agrégé quadrilatère (polygone à 4 sommets) en trois dimensions.
- b) Concevoir un algorithme réalisant les traitements suivants:
- Lecture au clavier d'un quadrilatère
 - Calcul puis affichage du périmètre de ce quadrilatère

Exercice n°9

- a) Concevoir un type agrégé "temps" de stockage d'une heure représentée par un nombre d'heures, de minutes et de secondes.
- b) Concevoir un algorithme réalisant les traitements suivants:
- Lecture au clavier d'un temps t
 - Lecture au clavier d'un nombre de secondes nbs
 - Modification du temps t par incrémentation de nbs secondes
 - Affichage écran de t

Exercice n°10

- a) Concevoir un type agrégé "localisation géographique" avec nombres de degrés en entier, de minutes en entier et de secondes en réel.

Exemple: Horloge astronomique de la cathédrale de Besançon: Longitude 6°01'49.08", latitude 47°14'00.96"

- b) Concevoir un algorithme permettant de convertir une localisation géographique en un nombre de degrés en réel (lecture au clavier de la localisation géographique, conversion et affichage du résultat de conversion). Concevoir un algorithme permettant de convertir un nombre de degrés en réel en une localisation géographique (lecture au clavier du nombre de degrés réel, conversion et affichage de la localisation géographique).
- c) Concevoir un type agrégé "localisation GPS" avec longitude et latitude.
- d) Concevoir un algorithme de calcul de la distance "à vol d'oiseau" existant entre deux localisations GPS définies à l'altitude 0.0.

Le rayon terrestre est de 6378.137 km. La formule de calcul est la suivante:

$$dlo = (lo2 - lo1) / 2.0;$$

$$dla = (la2 - la1) / 2.0;$$

$$a = \sin(dla)^2 + \cos(la1) * \cos(la2) * \sin(dlo)^2;$$

$$d = 2.0 * \text{atan2}(\text{sqrt}(a), \text{sqrt}(1.0 - a)) * \text{rayonTerrestre};$$

Où lo1, lo2, la1 et la2 sont les longitudes et latitudes en réel des localisations GPS.

Auteur: Nicolas JANEY
UFR Sciences et Techniques
Université de Besançon
16 Route de Gray, 25030 Besançon
nicolas.janey@univ-fcomte.fr
