**Choisir le type et la structure de données adéquats**

1. **Description**

Dans ce quiz, vous allez aider un professeur à entrer les notes de ses 30 élèves à l’aide d’un algorithme.  
Au fur et à mesure des questions, vous allez construire pas à pas cet algorithme en découpant un problème en petits sous-problèmes.  
Vous vous souvenez de cette pratique ? Alors c’est parti et bon courage à vous !

* **Question 1**

**Le professeur souhaite entrer le nom de chaque étudiant dans son programme.**

**Quel type de variable faut-il utiliser ?**

* + Un nombre entier
  + Un booléen
  + Une chaîne de caractères

*Une variable de type “nombre entier” permet d’enregistrer l’ensemble des nombres entiers, que ce soit un nombre négatif ou positif, comme -30, 20 ou 100. Néanmoins, ces types de variables ne permettent pas de stocker des nombres à virgule.*

*Un booléen est une variable qui prend au maximum 2 valeurs,  Vrai  ou  Faux  .*

*Une chaîne de caractères permet de stocker soit une lettre soit un ensemble de symboles, que ce soient des lettres, des espaces ou des caractères spéciaux. Dans le cas du nom d’une personne, il faut bien sûr choisir ce type de variable.*

* **Question 2**

**Le professeur va devoir entrer plusieurs informations concernant les étudiants. Pour cela, il a besoin  :**

* + d’entrer le **nom** de l’étudiant ;
  + d’enregistrer ensuite sa **présence à l’examen** ;
  + de saisir une **note** comprise entre 0 et 20.

**Attention, il est possible d’avoir des demi-points : une note de 15,5 est possible.**

**Quelles déclarations de variables permettent d’enregistrer les informations d’un étudiant ?**

* + nom ← "" : CHAÎNE DE CARACTÈRES
  + note ← 0 : NOMBRE DECIMAL
  + est\_présent ← "FAUX" : CHAÎNE DE CARACTÈRES
  + nom ← "" : CHAÎNE DE CARACTÈRES
  + note ← 0.0 : NOMBRE DECIMAL
  + est\_présent ← FAUX : BOOLÉEN
  + ok
  + nom ← "" : CHAÎNE DE CARACTÈRES
  + note ← 0 : NOMBRE ENTIER
  + est\_présent ← FAUX : BOOLÉEN
  + *Le nom contient des****lettres****, il faut donc forcément utiliser une****chaîne de caractère****pour enregistrer cette information.*
  + *Pour savoir si l’étudiant était présent, nous pouvons utiliser un****booléen****, car ce type permet d’enregistrer****deux états****, Vrai  ou  Faux  . Donc, dans notre cas, nous considérons l’état  Vrai  dans le cas où l’étudiant est présent, sinon  Faux  s’il est absent.*
  + *La note devra être enregistrée dans une variable de type «****nombre décimal****», car si vous aviez choisi un nombre entier, il manquerait l’information derrière la virgule. L’étudiant ne serait pas content.*
* **Question 3**

**À présent, nous voulons vérifier que le professeur n’a pas entré par erreur une note qui n’est pas comprise entre 0 et 20 inclus. Si c'est le cas, nous allons lui demander d’entrer à nouveau l’ensemble des informations à l’aide de la fonction  remplir\_informations  , et afficher le message suivant : La note n’est pas entre 0 et 20.  Sinon, le programme affiche  La note est correcte.**

**Cette fonction  remplir\_informations  contient les instructions permettant au professeur d’entrer les informations pour un étudiant, et de les enregistrer dans les 3 variables précédentes.**

**Quelle structure conditionnelle permet de gérer cette erreur de frappe ?**

* + Si note >= 0.0 OU note <= 20.0 :
  + Afficher "La note est correcte."
  + Sinon
  + Afficher " La note n’est pas entre 0 et 20."
  + remplir\_informations()
  + Si note <= 0.0 ET note >= 20.0 :
  + Afficher "La note est correcte."
  + Sinon
  + Afficher " La note n’est pas entre 0 et 20."
  + remplir\_informations()
  + Si note >= 0.0 ET note <= 20.0 :
  + Afficher "La note est correcte."
  + Sinon
  + Afficher " La note n’est pas entre 0 et 20."
  + remplir\_informations()
  + ok

*Pour faire cette vérification, vous avez besoin d’une structure conditionnelle****if...else****. Il y a deux conditions à vérifier :*

* + *Si la note est supérieure ou égale à 0.0.*
  + *Si la note est inférieure ou égale à 20.0.*

*Cependant, ces deux conditions doivent être réunies pour valider notre note. Il faudra donc utiliser le mot-clé  ET  pour les combiner. Ainsi,****si****ces deux conditions sont vérifiées, la note est valide,****sinon****il faut afficher le message d’erreur et appeler à nouveau la fonction   remplir\_informations  .*

* **Question 4**

**Précédemment, vous avez pu protéger la saisie de la note de l’élève grâce à notre structure conditionnelle. Effectivement, en cas d’erreur, le programme va demander une nouvelle fois d’entrer les informations et la note. Mais que se passe-t-il s’il se trompe encore ? Il ne se passe rien malheureusement, cette dernière note sera prise en compte comme note finale. Il nous faut donc une solution plus adaptée.**

**Nous allons cette fois utiliser une boucle !**

**Quelle boucle permet de remplacer la structure conditionnelle précédente pour gérer la saisie de la note ?**

* + Tant que note < 0.0 OU note > 20.0 :
  + Afficher " La note n’est pas entre 0 et 20."
  + remplir\_informations()
  + ok
  + Pour i allant de 1 jusqu’à 30 :
  + Afficher " La note n’est pas entre 0 et 20."
  + remplir\_informations()
  + Tant que note >= 0.0 OU note <= 20.0 :
  + Afficher " La note n’est pas entre 0 et 20."
  + remplir\_informations()

*Concernant le type de boucle, seule la boucle****Tant que****pourra nous aider à résoudre notre problème, car on ne connaît pas à l’avance le nombre d’itérations.*

*Ensuite, il ne faut pas se tromper sur la condition de boucle. Pour rappel, la condition doit être****VRAI****pour entrer et rester dans le bloc de la boucle. Ainsi, pour entrer dans la boucle et continuer à itérer, il faut que la note soit incorrecte.*

*Il y a deux conditions, il faut que la note soit soit strictement inférieure à 20, soit strictement supérieure à 0. En résumé : si la note est incorrecte, on entre dans la boucle et on y reste tant que la note n’est pas comprise entre 0 et 20 inclus.*

*Voici un tableau pour vous aider à comprendre la condition :*

|  |  |  |  |
| --- | --- | --- | --- |
| valeur | note < 0.0 | note > 20.0 | note < 0.0 **OU** note > 20.0 |
| -1.0 | VRAI | FAUX | VRAI |
| 30.0 | FAUX | VRAI | VRAI |
| 16.5 | FAUX | FAUX | FAUX |

* **Question 5**

**Vous avez pu terminer toutes les fonctions qui permettent de gérer les informations d’un étudiant. Mais il est maintenant temps de s’occuper des autres élèves. En effet, vous devez enregistrer les notes des 30 étudiants au même endroit grâce à un autre type de données : le tableau !**

**Quelle déclaration de variable correspond à notre tableau ?**

* + notes ← 0 : TABLEAU DE NOMBRES DECIMAUX
  + notes[30] : TABLEAU DE NOMBRES DECIMAUX
  + ok
  + notes ← 30.0: TABLEAU DE NOMBRES DECIMAUX

*Pour définir un tableau, il faut tout d’abord spécifier le****nom****du tableau, et ensuite mettre entre crochets la****taille****du tableau souhaité, c’est-à-dire le nombre de cases. Vous pouvez ensuite donner le****type****du tableau.*

* **Question 6**

**Vous avez précédemment créé un tableau qui permet de contenir 30 nombres entiers. Il va maintenant falloir le remplir à l’aide d’une boucle. En effet, chaque case contiendra la note d’un élève.**

**Quelle boucle est la plus adaptée pour inscrire les 30 notes d’élèves ?**

* + Pour i allant de 0 jusqu’à 29 :
  + Notes[i] ← note
  + ok
  + i ← 0
  + Tant que i < 30 :
  + Notes[i] ← note
  + i ← i + 1
  + Pour i allant de 1 jusqu’à 30 :
  + Notes[i] ← note

*On utilise une boucle****Tant que****lorsque le nombre d’itérations n’est pas connu à l’avance. La boucle****Tant que****n’est pas****adaptée****dans notre cas, car le nombre d’itérations est connu. Néanmoins, ce code fonctionne parfaitement.*

*Il faut donc utiliser une boucle****Pour****. Cependant, il faut faire attention à la valeur de l’indice  i  . Je vous rappelle que****l’indice****de la première case d’un tableau est****0****et non pas 1. Et de la même manière, l’indice de la dernière case est****29****dans notre problème. Il faut donc itérer avec un indice commençant à****0****et allant****jusqu’à 29****.*

* **Question 7**

**Mais nous avons oublié quelque chose non ? Comment savoir à qui appartient la note dans une case de notre tableau ?**

**Le professeur souhaite finalement associer un nom à une note.**

**Quel type de structure de données permet d’associer un nom et la note ?**

* + Une pile
  + Une file
  + Une table de hachage

*Dans les piles, il est uniquement possible de manipuler le dernier élément introduit dans la pile, on ne peut donc pas associer deux éléments.*

*Dans une file, on ajoute des éléments à une extrémité de la file, et on supprime des éléments à l'autre extrémité ; tout comme la pile, on ne peut pas associer deux éléments.*

*Une table de hachage permet d’associer deux éléments ; effectivement, la valeur sera associée à une clé unique qui est en quelque sorte l’indice de la case. Il faut donc choisir une table de hachage pour associer un nom à une note.*

* **Question 8**

**Finalement, le professeur a encore changé d’avis, il aimerait ordonner ses données sous la forme de nœuds interconnectés. Il souhaite avoir des nœuds liés au maximum avec 2 autres nœuds.**

**Quel type de structure de données pourrait-il choisir ?**

* + Arbre binaire
  + Arbre quelconque
  + Graphe

*Un****arbre quelconque****peut avoir des nœuds connectés avec****plus de deux****autres nœuds, tout comme le****graphe****.*

*Néanmoins concernant****l’arbre binaire****, comme son nom l’indique, un nœud peut être connecté****au maximum avec deux****autres nœuds. On dit qu’un parent peut avoir deux fils.*