

Exercices d'entraînement

Nom :

Exercice 1 – Le dégradé (5 points, 15 minutes)

Complétez les trous ci-dessous. On rappelle que **DessineLigne**(x_a, y_a, x_b, y_b, c) dessine une ligne allant du point d'abscisse x_a et d'ordonnée y_a au point d'abscisse x_b et d'ordonnée y_b , de couleur c .

Algorithme **DessineDegradé**

Entrées : _____ $x, y, largeur, hauteur, r1, r2, g1, g2, b1, b2$

Sortie : _____

Variables : _____ $tabR, tabG, tabB$; _____ i ; _____ $erreurAffichage$

Début

$tabR \leftarrow \text{tableauCouleurs}(r1, r2, largeur)$

$tabG \leftarrow \text{tableauCouleurs}(g1, g2, largeur)$

$tabB \leftarrow \text{tableauCouleurs}(b1, b2, largeur)$

_____ $\leftarrow \text{FAUX}$

Si $x+largeur > 800$ ou $x < 0$ ou $y < 0$ ou $y+hauteur > 600$ alors :

$ErreurAffichage \leftarrow \text{VRAI}$

Sinon :

_____ i de x à $x+largeur$ faire :

$\text{DessineLigne}(x, y, x+i, y+hauteur, \text{CouleurRGB}(\text{Case}(TabR, i-x+1), \text{Case}(TabG, i-x+1), \text{Case}(TabB, i-x+1)))$

Renvoyer $ErreurAffichage$

Fin

Algorithme **tableauCouleurs**

Entrées : _____ $x, y, longueur$

Sortie : tableau d'entiers

Variables : tableau d'entiers $tableau$; entier i

Début

$tableau \leftarrow \text{NouveauTableau}(longueur)$

$i \leftarrow 1$

_____ $i < longueur+1$ faire :

$\text{Case}(tableau, i) \leftarrow x + \text{PartieEntiere}(i * (y-x) / longueur)$

$i \leftarrow 1 + i$

_____ Renvoyer _____

Fin

Exercice 2 – Le bulletin de notes (12 points, 1h10)

Q1a. Écrivez un algorithme **minimum** qui étant donnés deux flottants a et b renvoie le minimum des deux.

Q1b. Que renvoie **minimum**(3,5) ? Que renvoie **minimum**(3.5,7) ?

Q2. Si un étudiant obtient une note finale inférieure à 2/20 dans un module à la fin de l'année, il doit redoubler. Nous allons donc, avec les questions Q2a à Q2c, écrire un algorithme pour détecter cela.

Q2a. Écrivez un algorithme **noteMinimum**, qui prend en entrée un tableau de notes d'un étudiant (un tableau de flottants, donc, contenant une note finale de module dans chaque case), et renvoie la note minimum qu'il a reçue parmi tous ses modules. Que renvoie **noteMinimum**({15,10,12,9,10.5}) ?

Q2b. Écrivez un algorithme **doitRedoubler** qui prend en entrée un tableau de notes d'un étudiant (de la même manière qu'à la question précédente), et renvoie un booléen qui indique s'il doit redoubler à cause de sa note minimum.

Q2c. Que renvoie **doitRedoubler**({5,10,12.5,13}) ? Que renvoie **doitRedoubler**({15,16,11}) ?

Q3. En fait, les tableaux de notes transmis par les enseignants contiennent deux notes pour chaque module : une note de compétences et une note de motivation. Nous allons donc, avec les questions Q3a à Q3d, écrire un algorithme pour faire la moyenne de ces deux notes dans chaque module, et renvoyer un tableau qui ne contient que la note finale de l'étudiant dans chaque module.

Q3a. Écrivez un algorithme **moyenne** qui prend en entrée deux flottants et renvoie la moyenne des deux.

Q3b. On cherche à écrire un algorithme **moyennesParModule**, qui prend en entrée un tableau dont chaque case correspond aux deux notes d'un module particulier, et renvoie un tableau dont chaque case contient la note finale de ce module. Ainsi, chaque case du tableau en entrée contient un tableau de 2 flottants : la note de compétences et la note de motivation, et chaque case du tableau en sortie contient la moyenne de ces deux notes. Quel est le type de la variable d'entrée de **moyennesParModule** ?

Q3c. Donnez un exemple d'utilisation de **moyennesParModule**. Vous devez donc écrire la phrase suivante en remplissant les trous : un appel de l'algorithme **moyennesParModule**(_____) renvoie le tableau ____.

Q3d. Écrivez l'algorithme **moyennesParModule**.

Exercice 3 – L'agrandissement de tableau (3 points, 5 minutes)

Q1. Écrivez un algorithme **AjouteCase** qui prend en entrée un tableau d'entiers *nombres* et un entier *a*, et renvoie en sortie un tableau d'entiers contenant une case de plus que *nombres*. Ce tableau doit avoir le même contenu que *nombres* dans toutes ses cases sauf sa dernière, et doit contenir *a* dans sa dernière case.