

M1.23.3 Algorithmique

Durée : 2h30

Documents non autorisés, calculatrices non autorisées.

Le barème est donné à titre indicatif.

Q1 Codage (5 points)

Coder dans *mesClasses\Ds2.java* la méthode suivante, puis la tester en modifiant et en utilisant : *tests\TestQ1.java*.

Fonction puissance(*x : entier*, *n : entier*):réel

var signe, cpt: *entier*

var resultat : *réel*

Début

 signe \leftarrow 0

Si *n*<0 **Alors**

 signe \leftarrow 1

n \leftarrow -*n*

Fin si

 cpt \leftarrow 1

 resultat \leftarrow 1

Tant que (cpt \leq *n*) **faire**

 resultat \leftarrow resultat **x*

 cpt \leftarrow cpt + 1

Fin tant que

Si signe=1 **Alors**

 resultat \leftarrow 1/ resultat

Fin si

renvoie(resultat)

Fin

Rem : ne pas hésiter à faire du copié colle, surtout pour les formules.

Q2 Conditionnelle (5 points)

Ecrire une méthode catégorie (dans *mesClasses\Ds2.java*) qui permet de savoir si une personne est dans une catégorie :

- entre 75 et 81 kg : poids mi lourds (code : 0)
- entre 81 et 91 kg : poids lourds (code : 1)
- plus de 91 kg : poids super lourds (code : 2)
- autre poids (code : -1)

Cette méthode prend un paramètre :

- un entier, le poids.

Elle retourne un *entier* qui code la catégorie

Le programme *tests\TestQ2.java* est déjà fourni.

Q3 Boucle (5 points)

Ecrire une méthode *fac* (dans *mesClasses\Ds2.java*) qui calcule la factorielle d'un nombre. La méthode prend en paramètre un nombre *entier* et retourne un *entier* qui est la factorielle du premier.

Pour rappel :

- $\text{fac}(0) = 1$
- $\text{fac}(1) = 1 * 1 = 1$
- $\text{fac}(2) = 2 * 1 = 2$
- $\text{fac}(3) = 3 * 2 = 6$
- $\text{fac}(4) = 4 * 6 = 24$
- $\text{fac}(n) = n * \text{fac}(n-1)$

Le programme *tests\testQ3.java* est déjà fourni.

Q4 Caisse enregistreuse (5 points)

Ecrire une méthode *caisse* (dans *mesClasses\Ds2.java*) qui simule un rendu de monnaie avec une caisse enregistreuse. Les nombres traités seront entiers.

Si un usager achète trois produits de 12, 20 et 5 Euros et ne dispose que d'un billet de 100 Euros. Votre programme devra lire la valeur du premier produit (12) puis celle du deuxième (20) puis celle du troisième (5) puis zéro qui signifie que la liste est finie, puis enfin pour finir l'argent versé (100). Fort de la valeur de l'argent versé et de la somme des produits achetés votre programme doit afficher le nombre de pièces de 5, de pièces de 2 et de pièces de 1 Euros à rendre. Dans notre exemple nous afficherons donc 12 pièces de 5, 1 pièce de 2 et une pièce de 1.

Cette méthode ne prend pas et ne retourne pas de paramètres.

Le programme *tests\testQ4.java* est déjà fourni.

Rem : je vous conseil de procéder en deux étapes la première est la lecture du prix, la deuxième est celle du calcul du rendu de monnaie.