Ecole Nationale des Sciences Appliquées - Al Hoceima

Département Mathématiques et Informatique

Travaux pratiques N°1 : Les bases du langage Java

Module : Programmation Orientée Objet en Java

Première Année Ingénierie des données

Année universitaire 2022/2023

Implémentation de quelques algorithmes simples

1. Ecrire un programme FormatHour qui prend en paramètre un nombre de secondes et qui permet de le formater en heures-minutes-secondes. Si on ne fournit pas un paramètre lors de l'appel du programme ou si le paramètre fourni n'est pas un entier, le programme devrait afficher message d'erreur.

   Voici un exemple d'exécution du programme :

   > java FormatHour 520

   8 minutes 40 secondes

   > java FormatHour

   > Arg Error.

   > java FormatHour 25217

   7 heures 17 secondes

2. Soit n un nombre naturel, un diviseur d positif de n est appelé diviseur propre si d est différent de n. Deux entiers naturels sont des nombres amis si chacun d'entre eux est égal à la somme des diviseurs propres de l'autre. Ecrire un programme Java qui affiche la liste des couples de nombres amis inférieurs à 20000.

3. Ecrire un programme Java qui calcule et affiche la valeur de π approchée sachant que $\frac{\pi^2}{6} = \sum_{i=1}^{\infty} 1/i^2$.

4. Une méthode qui calcule la valeur approchée de cos(x) en utilisant le développement en séries entières de la fonction cosinus :

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \cdots + (-1)^n \frac{x^{2n}}{(2n)!} + \cdots$$

La boucle de traitement devra s'arrêter quand la précision obtenue sera suffisante.

5. On considère deux suites numériques (Un) et (Vn) telles que pour n strictement supérieur à 2 :

$U_n = U_{n-1} + U_{n-2}$ Et $V_n = U_n/U_{n-1}$

La suite (Vn) tend vers une limite ℓ appelée nombre d'or. (ℓ = 1,6180339887498948482045868343656 4).

Ecrire un programme qui calcule une valeur approchée du nombre d'or avec une précision fixée.

6. On appelle nombres d'Armstrong les nombres entiers positifs tels que la somme des cubes de leurs chiffres est égale au nombre lui-même. Exemple : 153 est un nombre d'Armstrong. En effet : $1^3 + 5^3 + 3^3 = 153$

Écrire un programme JAVA qui affiche tous les nombres d'Armstrong inférieurs à 1000000. Résoudre cet exercice sans utiliser ni tableaux ni collections.

7. Ecrire une classe *Utils* qui fournit les méthodes utilitaires suivantes :
   - ✓ Méthode *factoriel(n)* qui calcul le factoriel d'un entier.
   - ✓ Méthode *factorielRecursif(n)* une version récursive de la fonction factoriel.
   - ✓ Méthode *combinaison* pour calculer C_m^n :

$$C_m^n = \frac{A_m^n}{n!} = \frac{m(m-1)\ldots(m-n+1)}{n!} = \frac{m!}{n!\,(m-n)!}$$

   - ✓ Méthode *PascalTriagle* qui affiche le triangle de Pascal (*https://www.bibmath.net/dico/index.php?action=affiche&quoi=./p/pascal.html*).
   - ✓ Ecrire un programme de test.

8- Écrivez un programme qui :

   a. demande à l'utilisateur de fournir un entier n ;
   b. crée un tableau de int à n cases et le remplit avec des valeurs demandées à l'utilisateur ;
   c. affiche la moyenne des valeurs fournies, ainsi que la plus petite et la plus grande de ces valeurs.
      - ✓ En utilisant la boucle *for*
      - ✓ En utilisant la boucle *foreach*

9- Écrire en Java les programmes suivants:

- Un programme qui fournit en retour un tableau constitué des n premiers nombres impairs, la valeur de n étant fournie en argument.
- Un programme qui reçoit en argument deux vecteurs d'entiers de même taille et qui fournit en retour un tableau représentant la somme de ces deux vecteurs.