

Mini-devoir séance 15

Note : Chaque exercice vaut 3 points pour un total de 21 points. Chaque jour de retard entraîne une pénalité de 3 points. Le premier jour est considéré terminé à 20H00, à la fin du TP. Il faut envoyer tous les programmes, de préférence dans une archive ZIP, à `dp.cnam@gmail.com`.

Exercice 1 Écrire un programme `Processing` qui permet de tracer 3 cercles tous centrés au pixel de coordonnées (300,400) sur une toile de taille 600×600 . Le rayon de chaque cercle doit être une valeur aléatoire, mais les cercles ne doivent pas toucher le bord de la toile.

Exercice 2 Écrire (sous java ou processing) une fonction `valeurPositive(int x)` qui renvoie `true` si $x \geq 0$ ou `false` sinon.

Exercice 3 Écrire une classe Java `Exo3` avec une méthode statique `conclureImpots(int revenus)` qui ne renvoie rien et qui affiche :

- pas d'impôts si le revenu est inférieur à 10000 ;
- des impôts si le revenu appartient à l'intervalle $[10000, 1.000.000]$
- vous avez besoin d'un parachute doré si le revenu dépasse 1.000.000 (euros).

Exercice 4 Écrire une classe Java `Exo4` avec une fonction `sommePairs(int n)` qui renvoie la somme des premiers n entiers pairs. Par exemple, pour $n = 5$ il faut renvoyer $2 + 4 + 6 + 8 + 10 = 30$. Il y a vraiment beaucoup d'approches possibles. N'hésitez pas à utiliser le bloc de code ci-après comme exemple.

```
int somme = 0;
for (int i=2; i <= ...; i=i+2) // attention : i=i+2
    somme = somme + i
```

Exercice 5 Écrire un programme `Exo5.java` avec une fonction boolean `tableauCroissant(int[] t)` qui renvoie `true` si le tableau `t` est trié de manière croissante ou `false` sinon. Par exemple, il faut renvoyer `true` pour `t={1,2,3,4}` ; ou `false` pour `t={1,2,5,4}` ;

Exercice 6 Écrire une classe `Exo6` avec une fonction double `noteFinale(double tp, double exam)` renvoie la note finale calculée avec les règles suivantes :

- (1) La note finale vaut 0 si la note d'examen est inférieure à 10 (échec)
- (2) La note finale est la note d'examen si $\text{exam} \geq 10$ et $\text{exam} \geq \text{tp}$.
- (3) La note finale est la moyenne entre la note d'examen et la note de TP si on a $10 \leq \text{exam} < \text{tp}$.

Indication : vous n'avez pas à tester cette relation/condition explicitement. La fonction doit pouvoir arriver au point (3) *uniquement si* elle n'appelle pas `return` au points (1) ou (2). Si elle arrive au point (3), alors la condition $10 \leq \text{exam} < \text{tp}$ est sûrement vérifiée. La fonction peut s'écrire en 5-7 lignes!!

Exercice 7 Continuer le programme précédent, en utilisant "copier-coller" pour copier la fonction `noteFinale` dans le nouveau programme. Modifier la méthode `main(...)` pour que le programme demande à l'utilisateur de saisir la valeur des deux notes (exam et TP) ; ensuite affichez la note finale. Le programme devrait être capable de gérer le cas où l'utilisateur ne saisit pas une valeur de type `double` acceptée par le langage. Par exemple, si l'utilisateur saisit « toto » ou « 6,223 », il faut re-demander la saisie (au moins une fois suffit). Il n'est pas obligatoire de tester que les notes représentent des valeurs de 0 à 20 ; par souci de simplification on imagine un monde qui permet toute valeur de note, y compris -1, -2, 21, etc.