



## Challenge : les algorithmes

### Résumé

Avez-vous compris la découpe en algorithmes, les exceptions et la javadoc ? Voyons ça en relevant le défi du "pas du zébu" :

Voici quelques conseils pour vous guider dans la résolution de tels problèmes :

- il convient d'abord de bien comprendre le problème posé ; assurez-vous qu'il est parfaitement spécifié ;
- résolvez le problème via quelques exemples précis ;
- mettez en évidence les variables « **données** », les variables « **résultats** » et les variables de travail ;
- n'hésitez pas à faire une ébauche de résolution en français avant d'élaborer l'algorithme définitif pseudo-codé ;
- déclarez ensuite les variables (et leur type) qui interviennent dans chaque algorithme ; les noms des variables risquant de ne pas être suffisamment explicites.
- Écrivez la partie algorithmique **AVANT** de vous lancer dans la programmation en Java.

### Afficher le titre

Écrivez un algorithme qui reçoit un nom en paramètre et qui affiche le texte :

DEV1 - Entête - Pas du Zébu - votre nom

### Longueur du pas du Zébu

La longueur du pas du Zébu est liée à son poids et au nombre PI par la formule (complètement farfelue) suivante :  $\text{pas} = \sqrt{(\text{poids} / |1 - \text{PI}|)}$   
Écrivez un algorithme qui reçoit le poids du Zébu en paramètre et retourne la longueur du pas du Zébu correspondante.

Si le poids reçu en paramètre est négatif, vous lancerez une erreur qui arrêtera le programme.

### La longueur du pas du Zébu au fil du temps

L'approximation du nombre PI a évolué au fil des ans.

- À l'époque de Babylone, il était approché par la valeur 3.125
- alors qu'en Égypte, il valait 3.160.
- Archimède, quant à lui, lui donnait la valeur 3.14185
- et Fibonacci 3.1418.



Écrivez un algorithme qui permette à l'utilisateur de choisir laquelle de 4 approximations de PI utiliser. Votre algorithme affichera les différentes valeurs de PI et l'utilisateur choisira celle qui lui convient. L'utilisateur fera ce choix en entrant le numéro d'ordre dans la liste affichée des différentes valeurs de PI.

Modifiez votre algorithme qui calcule le pas du zébu en lui passant un paramètre supplémentaire : l'approximation de la valeur de PI à utiliser.

Écrivez le code java correspondant avec la documentation javadoc pour chaque méthode. N'oubliez pas d'écrire la méthode `main` qui fera appel à votre méthode.

Aide en Java : la constante PI est définie dans la classe Math et vaut `Math.PI`