

S01 : Haskell (Introduction)

Enseignant : Stéphane LE PEUTREC

Assistant : Jonathan LAUPER

Instructions

- Deadline : jeudi suivant à 11:00
- Travail à rendre : un fichier zip de nom "nom_série_votre_nom_votre_prénom.zip" (exemple : SP1_Martin_Paul.zip). Cette archive doit contenir un seul dossier de nom "nom_série_votre_nom_votre_prénom". Ce dossier contient les fichiers suivants :
 - Un fichier imprimable rassemblant toutes vos réponses aux questions et vos solutions aux exercices (formats acceptés : pdf, txt, hmtl, rtf)
 - Les sources de vos solutions
- Ecrivez votre nom complet dans l'entête de tous vos fichiers

1. GHC Platform

Installez l'environnement de programmation GHC Platform sur votre ordinateur :
<http://www.haskell.org/platform/>

2. GHCi + tutoriel

a. Prenez connaissance des diverses commandes prédéfinies, (accessibles depuis la commande `:help`), essayez entre autres la commande `:t` et la commande `:set prompt`

- `:t (==)`
- `:i (==)`
- `:t max`

b. Faites le tutoriel qui peut être trouvé à l'adresse:

http://www.haskell.org/haskellwiki/Learn_Haskell_in_10_minutes

3. Déclaration de fonctions

Développez les fonctions qui suivent

- a) Une fonction qui calcule la somme de 4 entiers passés en paramètre
- b) Une fonction `max3` qui calcule la valeur max de 3 entiers passés en paramètres. Proposez 6 versions de cette fonction dont au moins une utilisant `if`, au moins une utilisant les gardes et au moins une utilisant `case` et au moins une utilisant `where`. Vous pouvez ou non utiliser la fonction prédéfinie `max`.
- c) Une fonction `signe` qui prend un nombre en paramètre et retourne "ce nombre est positif" si le nombre est positif, "ce nombre est négatif" si le nombre est négatif et "ce nombre est null" si le nombre est égale à 0. Proposez une version avec `if`, une version avec garde, une version avec `case`. Note : L'opérateur `++` permet de concaténer des chaînes de caractères.