

1ο Εργαστήριο
(ομάδες 1,2,3)

Στο τέλος του εργαστηρίου θα πρέπει να υποβάλλετε με `turnin` το αρχείο με τα προγράμματα σας, γράφοντας:

`turnin XX@myy401 filename`

όπου στη θέση του `XX` γράφετε την ομάδα στην οποία ανήχετε (01, 02 ή 03) και στη θέση του `filename` το όνομα του αρχείου που περιέχει τα προγράμματα Haskell τα οποία γράψατε.

Ασκηση 1.

Υλοποιήστε την παρακάτω συνάρτηση `f` σε Haskell:

$$f(x, y) = \begin{cases} \lfloor x^2 - 2xy^2 + 5y - 3 \rfloor & \text{αν } |x| < 1.2 \text{ ή } \sin x > 0.5 \\ \lceil (\sqrt{x^2 - \cos y})^{1/y} \rceil & \text{αλλιώς} \end{cases}$$

Μπορείτε να χρησιμοποιήσετε `if-then-else`, ωστόσο δεν επιτρέπεται να χρησιμοποιήσετε συνθήκες φρουρούς. Ο τύπος της συνάρτησης `f` θα πρέπει να είναι `Float -> Float -> Integer`

Υπόδειξη: Η Haskell έχει τις συναρτήσεις `abs`, `sqrt`, `cos`, `sin`, `floor` και `ceiling` που επιστρέφουν αντίστοιχα την απόλυτη τιμή, την τετραγωνική ρίζα, το συνημίτονο, το ημίτονο, το ακέραιο μέρος και την οροφή ενός πραγματικού αριθμού, καθώς και τους τελεστές `^` και `**` για ύψωση πραγματικού αριθμού αντίστοιχα σε ακέραια και πραγματική δύναμη.

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

```
Main> f 0.35 (-32)
-880
Main> f 21 2.3
227
Main> f 103.2 2
11
```

Ασκηση 2.

Γράψτε μία συνάρτηση `val` σε Haskell, η οποία θα υπολογίζει το γινόμενο των ψηφίων ενός θετικού ακεραίου, τα οποία είναι περιττοί αριθμοί. Αν κανένα ψηφίο του αριθμού δεν είναι περιττό, τότε η συνάρτηση θα πρέπει να επιστρέφει την τιμή 1. Μπορείτε να χρησιμοποιήσετε πρότυπα και συνθήκες φρουρούς, ωστόσο δεν επιτρέπεται να χρησιμοποιήσετε `if-then-else`. Ο τύπος της συνάρτησης `val` θα πρέπει να είναι `Integer -> Integer`

Υπόδειξη: Βασιστείτε στη συνάρτηση `sumOfDigits` του παραδείγματος 20, στη σελίδα 23 των σημειώσεων.

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

```
Main> val 2334567
```

```
315
```

```
Main> val 468
```

```
1
```

```
Main> val 5135799
```

```
42525
```