

- Please write all your Haskell functions from this exercise sheet into a single .hs-file and upload it in OLAT.
- You can use a template .hs-file that is provided on the proseminar page.
- The file should compile with ghci.
- Once the file has been uploaded, it cannot be changed or resubmitted!

Exercise 6.1 *Live exercise*

1. Implement a function which takes the price per unit of area `:: Integer` and the shape of the land `:: Shape` as input and calculates the area and the circumference altogether of a land and returns them in a pair where the first element (a pair itself) contains area and circumference `:: Double` and the second element is the price per unit of area of this land `:: Integer`. Note that `pi :: Double` is predefined in Haskell.

```
data Shape = Circle Double | Rect Double Double
land :: Shape -> Integer -> Pair (Pair Double Double) Integer
```

Implement another function `auction :: Pair (Pair Double Double) Integer -> Double` which takes the output of `land` as input and calculates the auction price of the land. Since people prefer less circumference, the auction price is calculated as:

$$P_{\text{auction}} = S_{\text{area}} * P_{\text{area}} + P_{\text{area}} * S_{\text{area}} / L_{\text{circum}} \quad (1)$$

How do you call these two functions sequentially?