All You Need Are Functions

Handout

SPGS 14 November 2016

Henrik Nilsson

Functional Programming Laboratory, School of Computer Science

University of Nottingham, UK

Try Haskell (1)

Point your browser to `http://tryhaskell.org`.

- A string in Haskell is the same as a list of characters. I.e.

  ```
  ['a', 'b', 'c'] = "abc"
  ```

  Try it: type in `['a', 'b', 'c']` to verify.
- Try functions `head`, `tail`, `reverse`, `sort` on your name. E.g. `head "Henrik"`. What do they do?
- Write an expression that extracts:
  - The second letter of your name
  - The last letter of your name

Try Haskell (2)

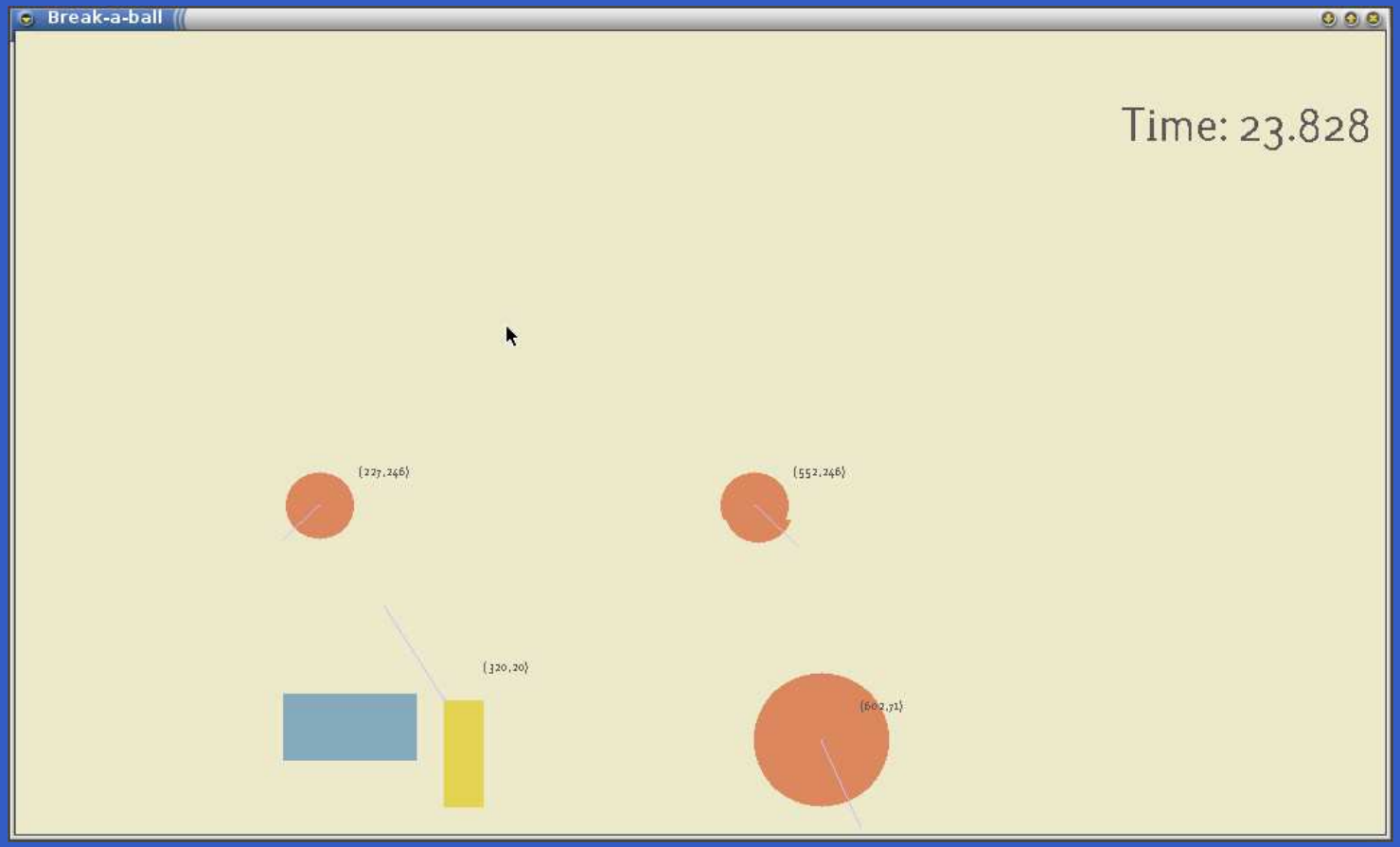
- What is `[1..10]`?
- Write an expression for the list of all integers from 50 to 100.
- Do `head`, `tail`, `reverse` work on lists of numbers?
- What is the type of `head`, `tail`, `reverse`?
  Hint: just type in e.g. `head` and hit return.
  What do the types mean?
- What does the function `sum` do to a list of numbers?
- Write an expression to sum all integers from 1 to 1000.

Try Haskell (3)

- `(*2)` is a function that multiplies a number by 2;
  `(^2)` is a function that squares a number. Try!
- `map` is a ***higher order*** function: it takes a function as an argument and applies it to every element in a list. Explain the result of:
  - map (*2) [1..10]
  - map (^2) [1..10]
- Sum the squares from 1 to 1000.
- What does `words` do to your full name?
- Extract the initials from your full name.

Take-home Game!

Download for free to your Android device!



Play Store: Pang-a-lambda (Keera Studios)

More information

- `http://www.haskell.org`
- John Hughes, recent retrospective: Why Functional Programming Matters
  `https://www.youtube.com/watch?v=FGQAP0GxlW8`