Programming Paradigms

Lab 6. Higher-order functions and ADTs

Outline

- Higher-order functions and lists in Haskell recap
- User-defined types in Haskell recap
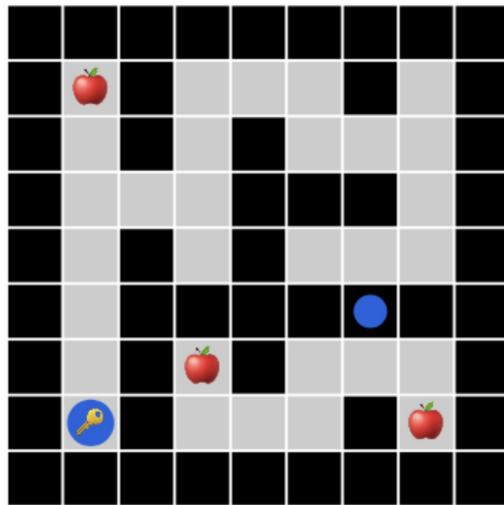- Exercise: N-body simulation

Rendering a tilemap

Exercise 6.1.

See prepared code snippet: https://code.world/haskell#PVe5pLviXrxL2FkIRKQ2SQw

Implement function **renderGrid** that renders a **Grid** of **Picture**s.

Try using higher-order functions.

Rendering a tilemap (with ADTs)

Exercise 6.2.
See prepared code snippet: https://code.world/haskell#PuRiYbTd_q_icJ2rAyaP6tw
Implement functions **renderItem**, **renderTile**.

```
-- | An item that can be placed on a floor tile.
data Item
  = Key DoorId   -- ^ A key for some door.
  | Coin         -- ^ A coin.

-- | A tile.
data Tile
  = Wall                   -- ^ A wall tile.
  | Floor (Maybe Item)     -- ^ A floor tile, possibly with some item on it.
  | Door DoorId            -- ^ A door (with its index).
```

Rendering a tilemap (with ADTs)

Exercise 6.3.
See prepared code snippet: https://code.world/haskell#PuRiYbTd_q_icJ2rAyaP6tw
Implement function **renderTileGrid**.

```
-- | An item that can be placed on a floor tile.
data Item
  = Key DoorId   -- ^ A key for some door.
  | Coin         -- ^ A coin.

-- | A tile.
data Tile
  = Wall                   -- ^ A wall tile.
  | Floor (Maybe Item)     -- ^ A floor tile, possibly with some item on it.
  | Door DoorId            -- ^ A door (with its index).
```

Modifying a tilemap (with ADTs)

Exercise 6.4.
See prepared code snippet: https://code.world/haskell#PuRiYbTd_q_icJ2rAyaP6tw
Implement function **removeItems** that removes items from all **Floor** tiles
in a **Grid** of **Tile**s.

Exercise 6.5.
Implement function **mapGrid** that applies a given function to every element in a **Grid**.

Exercise 6.6.
See prepared code snippet: https://code.world/haskell#PuRiYbTd_q_icJ2rAyaP6tw
Implement function **openDoors** that opens all **Door** tiles (i.e. replaces them
with **Floor** tiles) in a **Grid** of **Tile**s.

Playing with a tilemap

Exercise 6.7.

See prepared code snippet: https://code.world/haskell#PuRiYbTd_q_icJ2rAyaP6tw

Implement function `myTileGrid2` by converting `myCharGrid`.

Exercise 6.8. (*)

Use `activityOf` to make the game interactive.

Homework (self-study)

1. Install **Haskell** https://www.haskell.org/downloads/
2. Read **Learn you a Haskell for Great Good** Chapters 2, 4, and 5
   http://learnyouahaskell.com/chapters
3. Test yourself by implementing a program that renders a Koch snowflake of a
   given rank in Haskell on Code.World platform (https://code.world/haskell):