

Paradigmas de Programação

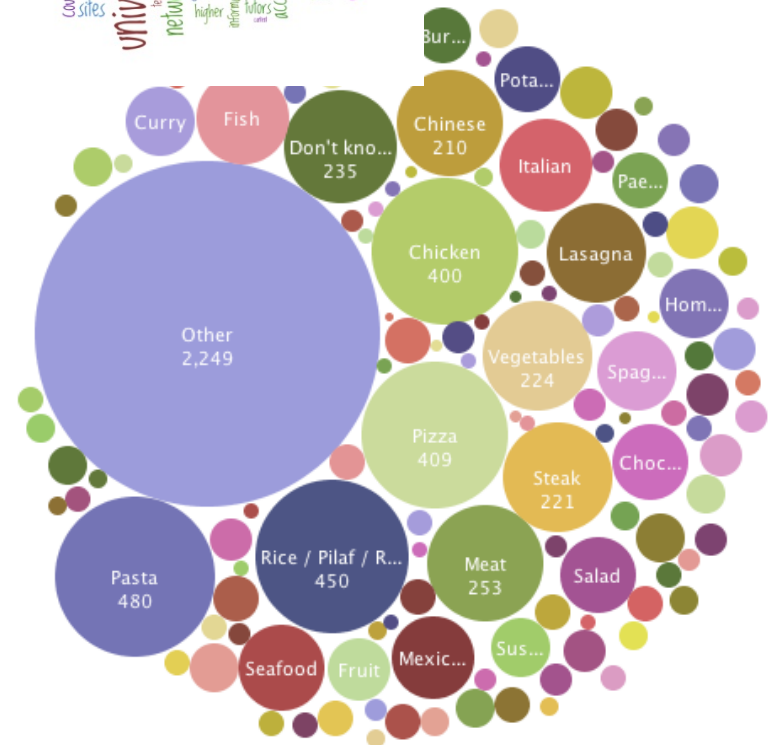
Linguagem Haskell

Prof^a Andréa Schwertner Charão

DELC/CT/UFSM

Tag cloud em Haskell

- Tag cloud é uma forma de visualização de dados
- A forma/cor de uma tag no gráfico é uma função de seu número de ocorrências



Algoritmo básico

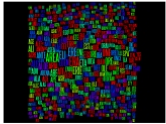
- Inspirado no wordle.net
- *"Participatory visualization with Wordle. F B Viègas, M Wattenberg, J Feinberg. IEEE Transactions on Visualization and Computer Graphics, 1137--1144, 2009."*

Wordle™ Home Create Gallery Credits News Forum FAQ Advanced

Wordle is a toy for generating “word clouds” from text that you provide. The clouds give greater prominence to words that appear more frequently in the source text. You can tweak your clouds with different fonts, layouts, and color schemes. The images you create with Wordle are yours to use however you like. You can print them out, or save them to the Wordle gallery to share with your friends.

[Create your own.](#)

View some examples created by others...

			
English notebook cover by Ace Acedemic! 3 years, 2 months ago	Period G by Meredith 3 years, 2 months ago	US Constitution by Jonathan 2 years, 11 months ago	Most Common Crossword Answers by Jonathan 3 years, 11 months ago

Algoritmo básico para círculos (bubbles)

frequências (peso) das tags => raios dos círculos

sort circle by radius, decreasing

for each radius:

 set initial circle position

 while circle intersects other circles:

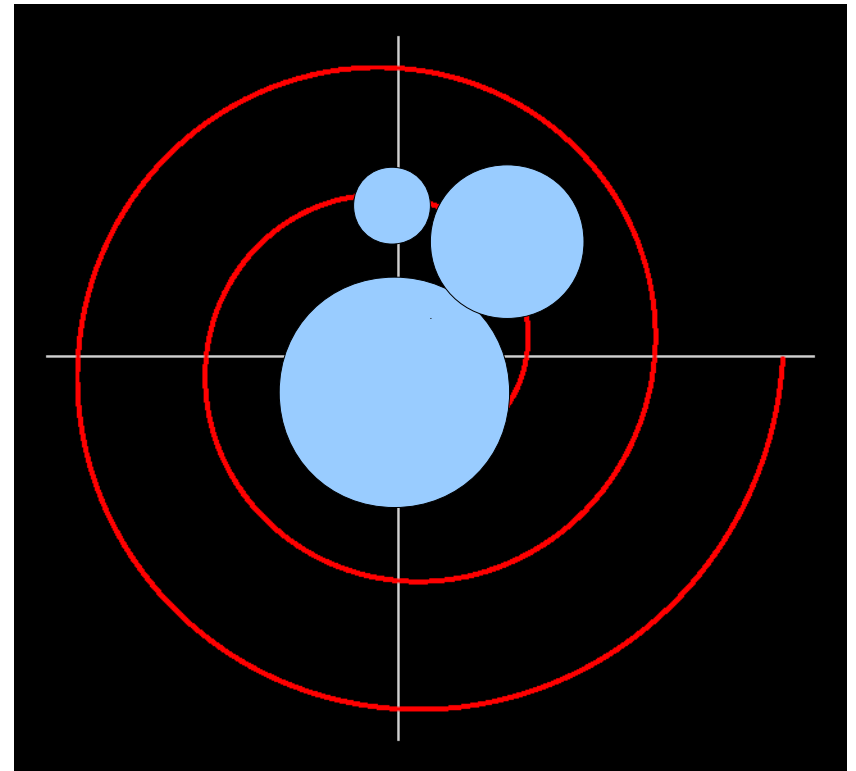
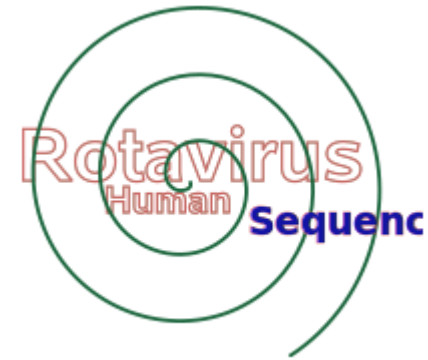
 update circle position

Update circle position

- Move o ponto em uma espiral a partir da posição inicial
- Equações paramétricas de uma espiral:

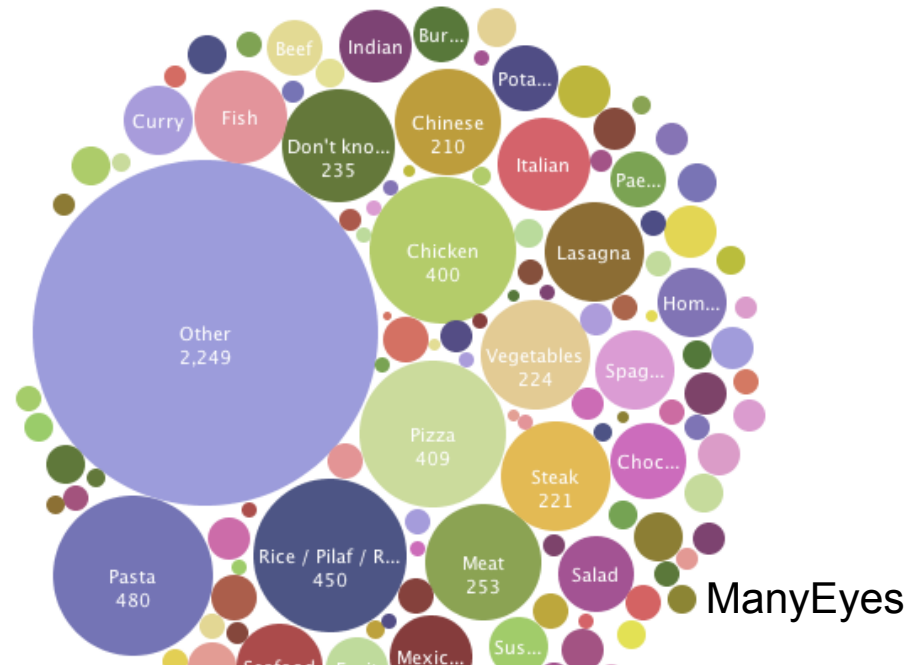
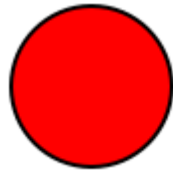
$$x = a * t * (\cos t)$$

$$y = a * t * (\sin t)$$



SVG (Scalable Vector Graphics)

```
<svg xmlns="http://www.w3.org/2000/svg" version="1.1">
  <circle cx="100" cy="50" r="40" stroke="black" stroke-width="2" fill="red"/>
</svg>
```



Simplificação: não é necessário colocar as tags dentro dos círculos
(é possível trabalhar somente com os pesos/frequências das tags)

Por onde começar?

Aproximações sucessivas:

- Programa que gera um SVG vazio (apenas o fundo)
- Programa que gera um SVG a partir de uma lista de círculos criada manualmente (fixa)
- Programa que gera círculos dispostos sobre uma espiral
- ...

Alguns recursos úteis

Tipos:

```
type Point      = (Float,Float)
type Color      = (Int,Int,Int)
type Circle     = (Point,Float)
```

Casamento de padrões:

```
circleArea :: Circle -> Float
circleArea (_, r) = pi*r^2
```

Definir funções auxiliares

- Calcular a distância entre 2 círculos
- Verificar se 2 círculos possuem intersecção
- Traduzir números de um intervalo para outro
- Deslocar um ponto
- Gerar strings com cabeçalho e conteúdo do SVG
- Aplicar transformações em listas
- Etc.