

Задание №11

Рекурсивное вычисление элементов списка на Haskell

I. Общая постановка задачи



На языке Haskell опишите реализацию функции `f11` с сигнатурой $(\text{Ord } a, \text{Fractional } a) \Rightarrow a \rightarrow a \rightarrow \text{Int} \rightarrow [a]$ для рекуррентного соотношения, соответствующего номеру вашего варианта. Запуск функции

```
f11 x eps n
```

должен приводить к вычислению для заданных чисел `eps` и `n` ($0 < \text{eps} < 1$) списка элементов $[y_0, y_1, \dots, y_m]$, где y_m — элемент заданной последовательности для такого минимального m , $0 < m < n$, при котором выполняется условие

$$|y_m - y_{m-1}| < \text{eps}.$$

Если такого элемента не найдётся, то функция должна выдавать $[y_0, y_1, \dots, y_n]$. При $n = 0$ функция должна выдавать $[y_0]$, независимо от значения `eps`.



Вспомогательные функции и значения должны определяться только в качестве локальных. Результат загрузки файла с решением в интерпретатор — только определение функции `f11`.



Не следует «вручную» довычислять элементы списка при его определении. Следует использовать только значения и соотношения представленные в задании.



В файле с программой приведите несколько вызовов функции `f11`, демонстрирующих корректную работу в различных ситуациях.



Файлу с программой дайте имя `task11-NN.hs`, где вместо `NN` — номер вашего варианта. Полученный файл загрузите на портал в качестве решения задания.



В функции ни одно выражение (подвыражение) не должно вычисляться дважды. В случае необходимости такого вычисления нужно связать значение вычисленного выражения с некоторым локальным именем для дальнейшего использования.



Реализация функции должна предполагать, что в ходе вызова параметры заданы корректно (не следует добавлять реализацию «защиты от дурака»).



Не следует делать предположений насчёт задания, не сформулированных явно в условии. Если возникают сомнения — задайте вопрос на форуме «Язык Haskell».

2. Пример выполнения задания

0.

$$y_0 = 1, \quad y_1 = \frac{x}{2}, \quad y_{n+1} = \frac{xy_n^2 - 5}{y_{n-1}y_n}$$

Решение: Содержимое файла task11-00.hs:

```
-- результат функции описывается как заданное количество элементов от списка lst
-- если в списке элементов меньше заданного числа, то take выделяет все
f11 x eps n = take (n + 1) lst
  where
    -- который, в свою очередь, начинается с двух заданных значений, а хвост
    -- его хвоста вычисляется как значение функции от этого же списка
    lst = 1.0 : x / 2.0 : elemLst lst
    -- функция предполагает, что список бесконечный и в нем можно выделить
    -- два первых элемента по которым можно вычислить очередной
    -- здесь y1 -- элемент списка перед тем, который нужно вычислить
    -- y0 -- элемент списка перед y1
    elemLst (y0 : ys @ (y1 : _)) =
      if abs (y0 - y1) < eps then [] -- выход по разнице между соседями
      -- иначе вычисляем очередной элемент, он будет головой результата
      -- хвостом результата будет значение функции от хвоста аргумента
      else (x * y1 * y1 - 5) / (y0 * y1) : elemLst ys

--тестовые запуски оформляем в виде функции main без аргументов
main = do -- тело функции -- последовательное выполнение
  let y1 = f11 0.3 0.001 500 -- связываем вызов с именем
  putStrLn "f11 0.3 0.001 500"
  print $ y1 !! 10
  print $ y1 !! 100
  print $ y1 !! 500
  let y2 = f11 0.5 0.001 500
  putStrLn "f11 0.5 0.001 500"
  print $ y2 !! 10
  print $ y2 !! 100
  -- 435-й элемент не вычислялся, т.к. разница между 433 и 434
  -- меньше второго аргумента функции
  print $ y2 !! 433
  print $ y2 !! 434
  -- print $ y2 !! 435
  let y3 = f11 1.7 0.001 500
  putStrLn "f11 1.7 0.001 500"
  print $ y3 !! 10
  print $ y3 !! 100
  print $ y3 !! 500
```

Текст примера можно загрузить с портала.

3. Необходимый минимум

Для выполнения работы потребуются сведения о следующих функциях, операциях и конструкциях:

- конструкция для определения функций
- конструкция `if ... then ... else ...`
- конструкция `... where ...`
- конструкторы списка `[,] , : , []`
- арифметические операции `+`, `-`, `*`, `/`
- функция нахождения абсолютной величины `abs`
- логические операции `||`, `&&`, `not`
- операции сравнения `<`, `>`, `==`, `/=`, `<=`, `>=`
- операция извлечения элемента из списка `!!` (только в тестах)
- функции для обработки списков `take`, `map`, `filter`, `zipWith`.
- тригонометрические функции `sin`, `cos`



Нельзя использовать конструкции и функции, не перечисленные в этом разделе (за исключением функций собственного сочинения). Если вы считаете, что для выполнения какого-то из заданий необходима функция/конструкция, отсутствующая в перечислении, то задайте вопрос на форуме «Язык Haskell»;

4. Варианты заданий

1. $y_0 = 1, \quad y_1 = x,$
 $y_{n+1} = \frac{y_{n-1} + 1}{n(y_{n-1} + 2)} y_n^2 + y_n x^2$

2. $y_0 = 4, \quad y_{n+1} = \frac{x - y_n^3}{20n}$

3. $y_0 = 0, \quad y_1 = 0.5,$
 $y_{n+1} = x \frac{y_n + y_{n-1}}{2} + y_{n-1} x^2$

4. $y_0 = 1, \quad y_1 = \frac{x}{2},$
 $y_{n+1} = 2y_n - \frac{x^2}{n^2} y_{n-1}$

5. $y_0 = 4.7, \quad y_{n+1} = n \sin y_n + x$

6. $y_0 = 0, \quad y_1 = x, \quad y_2 = \frac{x^2}{2}$
 $y_{n+1} = \frac{y_n - 2y_{n-2} + y_{n-1}}{y_n^2 - y_{n-2}^2}$

7. $y_0 = 1, \quad y_1 = x,$
 $y_{n+1} = \frac{x}{n} y_n^2 - y_{n-1}$

8. $y_0 = 0.1, \quad y_1 = 2 \cos x, \quad y_2 = \sin^2 x,$
 $y_{n+1} = \frac{n \sin y_n + x}{\cos y_{n-1} + \sin y_{n-2}}$

9. $y_0 = 1, \quad y_1 = x,$
 $y_{n+1} = \frac{x+7}{n x^2} y_n^3 - \frac{y_{n-1}}{n^2}$

10. $y_0 = 0.1, \quad y_{n+1} = \frac{1}{6} (x y_n^2 + 0.05)$

11. $y_0 = 1, \quad y_{n+1} = x y_n^4 + \frac{1}{n}$

12. $y_0 = 1.8, \quad y_{n+1} = \frac{3}{x} y_n - \frac{10}{13} y_n^2$

13. $y_0 = 1, \quad y_1 = \frac{x}{6},$
 $y_{n+1} = x^2 y_n + \frac{y_{n-1}}{n^2}$

14. $y_0 = 0.1, \quad y_{n+1} = \frac{y_n^3 + 0.05}{x}$

15. $y_0 = 0.03, \quad y_1 = 0.0001,$
 $y_{n+1} = \frac{x}{2n} y_n^2 + y_{n-1}$

16. $y_0 = 1, \quad y_1 = 0.5x,$
 $y_{n+1} = \frac{x^2 + 5}{100n} y_n + x y_{n-1}^3$

17. $y_0 = 1, \quad y_1 = \frac{x^2}{5},$
 $y_{n+1} = \frac{2x}{n^2} y_n + x^2 y_{n-1}$

18. $y_0 = 0, \quad y_1 = x^4,$
 $y_{n+1} = \frac{5x y_n}{n} - y_{n-1}$

19. $y_0 = 0, \quad y_1 = 1,$
 $y_{n+1} = \frac{n+5}{n^2} y_{n-1} + x y_n$

20. $y_0 = 1, \quad y_1 = x^2,$
 $y_{n+1} = \frac{5x y_n}{n^2} - y_{n-1}^2$

21. $y_0 = 1, \quad y_1 = \frac{x}{5},$
 $y_{n+1} = (x-1) \frac{y_n}{n} + y_{n-1}$

22. $y_0 = 0, \quad y_1 = x^2,$
 $y_{n+1} = x^2 y_n^2 + x \frac{n}{n^2 + 1} y_{n-1}^3$

23. $y_0 = 1.0, \quad y_1 = x^2, \quad y_2 = \frac{x^3}{2}$
 $y_{n+1} = \frac{x y_n + y_{n-2}^2}{n y_{n-1}}$

24. $y_0 = 0, \quad y_1 = x,$
 $y_{n+1} = \frac{x y_n + n y_{n-1}}{n+1}$

25. $y_0 = 2011,$
 $y_{n+1} = y_n \cdot \left(\frac{n+1}{n+2} \right)^2 + \frac{x}{(n+2)^2}$