

1. Напишите функцию которая получает на вход две экспериментально померенные зависимости  $(x_1, y_1, x_2, y_2)$ , аппроксимирует их прямыми методом наименьших квадратов и возвращает координаты пересечения. Воспользуйтесь функцией  $m1 = \text{lm}(y1 \sim x1)$ . Коэффициенты регрессионной прямой находятся в `l$coefficients`.
2. Напишите функцию генерирующую все возможные последовательности данной длины из данного алфавита (без использования функции `combn` и аналогичных)
3. Напишите функцию находящую все вхождения слова ( $w$ ) в текст ( $t$ ) с не более чем  $n$  ошибок.  $W$ ,  $t$  и  $n$  — параметры функции. Функция возвращает позиции начала вхождений и количества ошибок.
4. Напишите функцию которая сравнивает набор слов одинаковой длины и возвращает матрицу попарных расстояний — долю несовпадающих букв.
5. Напишите функцию генерирующую случайную ломанную (матрица с двумя колонками —  $x$  и  $y$ ). Длина шага и поворот определяется (от предыдущего направления) определяется случайно исходя из равномерного распределения. Диапазон значений длин и поворота, равно как и число шагов — параметры функции. Нарисуйте сгенерированную ломанную
6. Напишите функцию которая рассчитает траекторию луны относительно Солнца для заданного набора временных точек. Считайте, что Земля движется по окружности радиуса  $r_1$  вокруг Солнца с угловой скоростью  $v_1$ . Луна движется по окружности радиуса  $r_2$  вокруг Земли с угловой скоростью  $v_2$ .  $R_1$ ,  $r_2$ ,  $v_1$ ,  $v_2$  и  $t$  — вектор временных точек — аргументы функции. Выберите адекватные параметры и нарисуйте траекторию.
7. Пьяный идет по мосту шириной  $l$  шагов. Каждый шаг пьяный смещается случайно на один шаг в право или влево. Если пьяный переступит через край моста — он падает и умирает. Напишите функцию моделирующую движение пьяного вплоть до падения. Ширина моста и положение пьяного — начальные параметры. Нарисуйте на одном графике 100 разных траекторий.
8. Напишите функцию получающую на вход квадратное уравнение в текстовом виде и возвращающую его корни и экстремум (вектор длины 3, или 1 если корней нет): `solveQ = function(«10.3 + 16x — 18x2»)`.
9. Напишите функцию, которая добавляется к матрице «размазанное» значение. Параметрами функции являются матрица, координаты ячейки в матрице (строка и столбец) и значение. Функция должна добавить к каждой ячейке матрицы указанное значение делённое на единицу плюс сумма квадратов расстояния данной ячейки от ячейки указанной аргументами функции (расстояние — разница порядковых номеров столбцов/строк) в степени  $a$ , которая тоже является параметром функции. Добавьте несколько точек и нарисуйте матрицу при помощи функции `image`. Должно получиться что-то такое (добавлено 5 точек):



10. Напишите функцию проверяющую неравенство треугольника для всех троек элементов

из заданной матрицы попарных расстояний. Функция возвращает NULL если неравенство треугольника выполняется для всех троек или номера элементов из одной тройки, для которой неравенство не выполнено.

11. Напишите функцию которая интегрирует заданную функцию (аргумент функции) методом трапеций. Функция должна увеличивать число трапеций до тех пор, пока разница в вычисленном значении между соседними шагами не станет меньше наперед заданного числа (diff) или пока количество итераций не станет слишком большим. Пример: `integrate(FUN=function(x){x+x^2},from=0,to=1,diff=1e-10,max.iter=1e4)`.
12. Напишите функцию которая получает на вход набор из  $m$  точек в  $n$ -мерном пространстве (в виде матрицы  $m$  строк на  $n$  столбцов) и находит все пары точек находящиеся на расстоянии менее заданного порога (аргумент функции) без использования функции `dist`.
13. Напишите функцию моделирующую движение грузика подвешенного на пружинке в невесомости. На грузик действует сила упругости пружинки ( $-x \cdot k$ , где  $x$  — смещение а  $k$  — коэффициент упругости) и сила сопротивления среды ( $-v \cdot f$ , где  $v$  — скорость, а  $f$  — коэффициент сопротивления среды). Функция моделирует движение временными шагами постоянной протяжённости  $t$ . Функция возвращает положение и скорость в каждый момент времени. Начальное смещение  $x$ ,  $k$ ,  $f$ ,  $t$  и  $n$  — число шагов — параметры функции. Нарисуйте зависимость положения груза от времени.
14. В ряду чисел следующее значение является суммой кубов цифр предыдущего числа. Напишите функцию считающую следующий член ряда на основании данного. Напишите функцию считающую данный ряд от любого числа вплоть до сходимости (появления повторов). Сколько и каких финальных состояний есть у таких рядов начинающихся с чисел от 1 до 1000?
15. Напишите функцию генерирующую все простые числа меньше некоторого заданного числа при помощи решета Эратосфена: перебираете все числа от 1, проверяете делится ли оно без остатка на хоть одно из простых чисел которые вы уже нашли, если нет — добавляете его в список простых чисел.
16. Напишите функцию открывающую скобки в выражении  $(a+b)^n$ , функция должна, брать на вход  $n$  и возвращать текст:  
`fun(3)`  
`«a^3+3a^2b+3ab^2+b^3»`
17. Напишите функцию рисующую текстом барплот для заданного вектора (вертикально). Максимальная высота, символ для отрисовки и разделитель между колонками — параметры функции  
`textBarplot(c(1,3,2),pch="*",sep=" ",max.height=6)`  

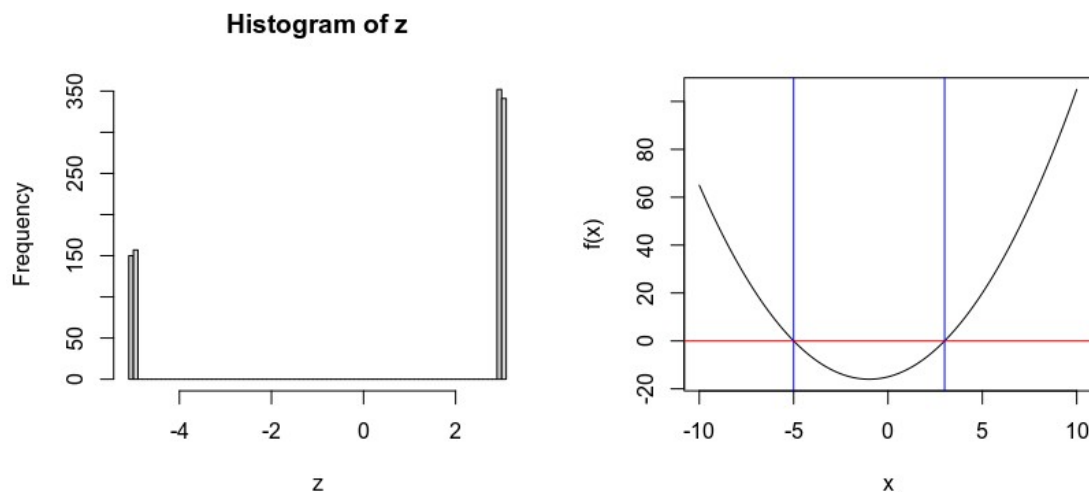
```

      *
      *
     * *
     * *
    * * *
    * * *

```
18. Напишите функцию проверяющую кто выиграл в игре в крестики-нолики. Функция получает на вход матрицу 3 на 3 заполненную 0 и +, возвращает NA (ничья) 0 или +.
19. Напишите функцию находящую элемент в отсортированном векторе методом деления пополам (аналог функции `findInterval`, ее использовать нельзя)
20. Напишите функцию генерирующую первые  $n$  членов ряда Фибоначи. Модифицируйте функцию так, а) первые  $k$  членов и б) количество последних членов ряда суммируемых на каждой стадии были параметром функции.

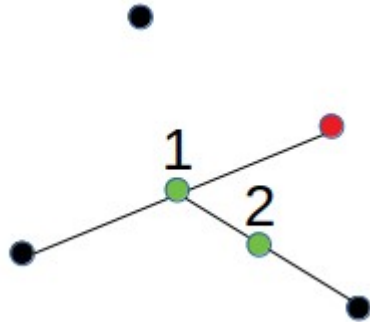
21. Напишите функцию получающую на вход текстовое представление полинома и возвращающая текстовое же представление его производной:  
`derivative(«1+10*x^3+7*x^5»)`  
`«30*x^2+35*x^4»`
22. Напишите простой генетический алгоритм для поиска корней произвольных функций одного аргумента. Алгоритм берет на вход функцию, диапазон в котором надо искать корни, количество поколений, размер популяции долю генотипов которые надо выкинуть, частоту мутаций и пределы размера мутаций (в долях от диапазона исходных значений). Популяция иницируется случайными значениями из диапазона. Каждый цикл для каждой особи вычисляется значение функций, особи сортируются по абсолютному значению, и указанная доля с максимальными значениями удаляется. Освободившиеся места заполняются копиями случайно выбранных оставшихся особей. Далее значения указанной части особей подвергаются мутации — к ним добавляются случайные нормально распределенные значения со средним равным нулю и стандартным отклонением равным указанной доле диапазона. Функция возвращает итоговую популяцию. Протестируйте на  $f = \text{function}(x) \{-15 + 2*x + x^2\}$ , должно получиться примерно так:

```
z=genAlg(f, from=-10, to=10, N=1000, generations = 100, selection = 0.2, mutation = 1e-3, mfreq = 0.5)
par(mfrow=c(1,2))
hist(z, 100)
x = seq(-10, 10, length.out = 100)
plot(x, f(x), t='l')
abline(h=0, col='red') /
abline(v=mean(z[z<0]), col='blue')
abline(v=mean(z[z>0]), col='blue')
```



23. Напишите функцию для генерации сиракузской последовательности ( $n_{i+1} = n_i/2$  если  $n_i$  четное и  $n_i*3+1$  если не четное) и проверьте при помощи нее [гипотезу Коллатца](#) (что все такие ряды сходятся к 1) для чисел от 1 до 10000. Нарисуйте зависимость длины ряда до достижения единицы от начального значения.
24. Выполните:
1. Создайте три случайные точки на плоскости  $[0,1]*[0,1]$ , нарисуйте их черным цветом
  2. Создайте еще одну точку и нарисуйте ее красным
  3. Создайте  $N$  ( $\sim 1e5$ ) точек следующим алгоритмом:
    - a) Красная точка – стартовая
    - b) Выберите случайно одну точку из трех созданных в шаге 1
    - c) Создайте точку являющейся серединой между стартовой точкой и точкой

- выбранной на стадии b
- d) Перенесите стартовую точку в точку созданную на шаге с
  - e) Повторите b-d N раз
4. Нарисуйте все получившиеся точки другим цветом и символом ‘.’.



\* «создайте точку» значит запишите ее координаты в переменную. Рисовать можно в самом конце