Заликов Руслан

v 0.2.3 – программа работает за (k + 6)n + (k + 1)n^2 где n количество точек а k рандомное число так как оно ровно попытки с которой алгоритм смог подобрать такие n точек чтобы они не накладывались друг на друга. При ошибке, что какая то точка наложилась, рандом начинает заново подбирать, поэтому k [1 , +) следовательно сложность алгоритма равна O(k), и если избавится от k то сложность станет O(n^2) что тоже не самое лучшее решение, так что после надо будет избавляться от n^2

Цель на v 0.2.4:

Избавиться от k

v 0.2.4 – Избавился от k, теперь задается не n случайных пар чисел и проверяется друг с другом, а весь экран делится на n участков, и у для каждой кнопки на участке i выбирается рандомная точка х от левой границы + радиус до правой границе – радиус, а по у просто рандомное значение от верха экрана + радиус до нижнего – радиус потому что в pygame счет по y идет сверху вниз, теперь функция которая работала за 2n + k(n^2 + n) работает за 2n

теперь сложность кода О(n^2)

исправлен вывод результата, теперь выводится моментально после прохода раунда

Цель на v 0.2.5:

Избавиться от n^2

V 0.2.5 – Пофиксил направление шаров, раньше они могли появляться со скорость 0, 0, теперь же такого произойти не может, и статичные шары не будут спавниться. Поставил задержку после попадания, чтобы пользователь успевал понять что происходит вокруг

Так же удалось в функции, которая обновляет картинку и соответственно перемещает точки, уменьшить количество проверок с n^2 до n(n+1)/2 хоть и порядок сложности все равно квадрат, все равно прогресс очевидно виден

Цель на v 0.2.6: уменьшить сложность еще, есть пару идей, протестировать, и реализовать код который уменьшит сложность до n(n-1)/2