

*Дедлайн. 20.10 23:59:59***Задача 1 (1 балл). Зацикленный**

Вам дан указатель на начало односвязного списка. Приведите алгоритм, позволяющий за линейное время определить, есть ли в нем цикл: если да, то какая его длина и узел из цикла.

Задача 2 (2 балл). Палкой по гоблину

Есть N гоблинов, у каждого h_i очков здоровья. У вас есть магический посох, позволяющий за один удар по i -му гоблину снести ему p очков здоровья и q всем остальным. Приведите алгоритм поиска минимального числа ударов, чтобы перебить всех гоблинов, работающий за $\mathcal{O}(N \log \max h_i)$ времени.

Задача 3 (2 балла). Прибавляй, арифметизируй!

Дан массив a_1, \dots, a_N . За один запрос (l, r, b, d) прибавляется арифметическая прогрессия на подотрезке, то есть

$$\begin{aligned} a_l &\rightarrow a_l + b \\ a_{l+1} &\rightarrow a_{l+1} + b + d \\ a_{l+2} &\rightarrow a_{l+2} + b + 2d \\ &\vdots \\ a_{l+k} &\rightarrow a_{l+k} + b + (k-1)d \\ &\vdots \\ a_r &\rightarrow a_r + b + (r-l-1)d \end{aligned}$$

Выведите массив после Q запросов. Время работы: $\mathcal{O}(N + Q)$.

Задача 4 (2 балла). В поисках сдвига

Отсортированный массив a_1, \dots, a_n сдвинули циклически на k вправо. Найдите это k .

- (1 балл) Если все элементы различны, то за $\mathcal{O}(\log n)$ времени.
- (1 балл) Докажите, что если могут быть одинаковые элементы, то любой алгоритм поиска k работает за $\Omega(n)$.

За каждый пункт по 1 баллу.

Задача 5 (2 балла). В поисках подгорода

На огороде растут помидоры и огурцы. Огород прямоугольный и поделен на квадратные сектора 1×1 метр. На каждом секторе растут либо огурцы, либо помидоры. Найдите самый большой прямоугольный участок, на котором растут только помидоры.

Указание. Сначала научитесь находить ближайший меньший слева и справа элементы.

Задача 6 ($3 + 2^*$ балла). Деамортизация

Деамортизируйте динамический массив!

На лекции вам приводили динамический массив с реаллокацией в 2 раза при попытке добавить в конец элемент, когда память закончилась. Операция добавления в конец `push` в той парадигме работала за *амортизированное* $O(1)$. Вам предлагается разработать такой динамический массив, в котором

1. `push` работает за чистые $O(1)$. Напомним, что выделение памяти или аллокация и очистка памяти (деаллокация) работает за $O(1)$ в нашем курсе. Выделенная память или `capacity` должна быть $\Theta(size)$, где `size` — число элементов в массиве.
2. Первый пункт и еще допускается функция `pop` — удаление из конца. Требование на `capacity = \Theta(size)` должно сохраняться, то есть после большого числа вызовов `pop` массив должен уметь адаптивно сжиматься.

Джависты, у вас в этой задаче нет сборщика мусора, придется самим управлять памятью: вызывать запросы получения и отдачи памяти ядру операционной системы.

Задача 7 (3 балла). k -ичная куча куч.

Рассмотрим k -ичную кучу. Это куча, где у узла k детей (считайте, что у нее массив детей) кроме, быть может, крайнего узла.

- Выведите формулу для индекса i -го ребенка j -го узла. Выведите формулу для родителя i -го узла.
- Оцените время работы просеиваний в такой куче.
- Пусть дети хранятся не в массиве, а в бинарной куче. Оцените время работы просеиваний.

За каждый пункт по одному баллу.