1 Алго ревью. Футбольная команда

1.1 Описание алгоритма

1) Сортировка по эффективностям игроков, QuickSort
2) Проход массиву двумя указателями и вычисление максимальной суммарной эффективности
3) Сортировка индексов найденных игроков, QuickSort

1.2 Доказательство корректности

Отсортируем игроков по эффективности и пронумеруем их $\{1, 2, \ldots, n\}$. Обозначим подмножество индексов выбранных игроков для команды, как $S = \{i_1, i_2, \ldots, i_k\} \in \{1, 2, \ldots, n\}$, причем $i_1 < i_2 < \ldots < i_k$. $E-$ массив эффективностей игроков. Тогда $E[i_1] \leqslant E[i_2] \leqslant \ldots \leqslant E[i_k]$. Случай $|S| = 1$ всегда подходит. Будем считать, что пробуем найти команду размера $|S| \geqslant 2$. Если выполнено условие $(*)$:

$$E[i_k] \leqslant E[i_1] + E[i_2], (*)$$

$$\Rightarrow E[w] \leqslant E[i_k] \leqslant E[i_1] + E[i_2] \leqslant E[u] + E[v], \forall u, v, w \in S$$

$$\Rightarrow E[w] \leqslant E[u] + E[v]$$

Значит условие $(*)$ - необходимо и достаточно для сплоченности команды.

Лемма 1. *Стоит рассматривать только команды, индексы которых образуют отрезок в отсортированном массиве.*

Доказательство *Для любого множества* $S = \{i_1, i_2, \ldots, i_k\}$ *можно сопоставить множество* $S' = \{i_k - (k-1), i_k - (k-2), \ldots, i_k\}$, *с такими свойствами:*

- $\forall j \in [1, k], i_j < i_{j+1} < \ldots < i_k$
  $i_j + (k-j) \leqslant i_k \Rightarrow i_j \leqslant i_k - (k-j)$
  Тогда $E[i_j] \leqslant E[i_k - (k-j)] \Rightarrow \sum_{i \in S} E[i] \leqslant \sum_{i \in S'} E[i]$

- $E[i_k] \leqslant E[i_1] + E[i_2] \leqslant E[i_k - (k-1)] + E[i_k - (k-2)]$
  $\Rightarrow E[i_k] \leqslant E[i_k - (k-1)] + E[i_k - (k-2)] \Rightarrow (*) \Rightarrow S'$ *- сплочены*

Значит мы нашли множество индексов S' который составляет отрезок, который оказался сплоченным и суммарная эффективность которого не меньше S. ■

Будем пробегать по отсортированному массиву двумя указателями, индексом i, и хранить левую границу отрезка $last$ и текущую сумму эффективностей. Изначально $i = last = 1$, и мы уже нашли сплоченную команду. Каждый раз после увеличения i, будем увеличивать $last$ до тех пор пока команда не станет сплоченной по условию $(*)$. Такой $last$ всегда найдется, как минимум есть $last := i$, команда размера $|S| = 1$. И надо не забыть пересчитывать суммарную эффективность команды отнимая эффективность убранных игроков, и прибавляя эффективность добавленных игроков, и пересчитывать максимальный ответ.

Лемма 2. *Подмножество сплоченного множества тоже сплочено.*

Лемма 3. *Дополнение не сплоченного множества тоже не сплочено.*

Пусть при переборе двумя указателями мы находимся в состоянии сплоченного отрезка $[j, i]$ и хотим двинуться в $i + 1$ и найти для него левую границу j'. Из первичности j для индекса i вытекает, что $\forall j' < j, [j', i]$ - не сплочен. Тогда $[j', i + 1]$ тоже не сплочен, имеет смысл перебирать только $j' \in [j, i + 1]$. Значит просто идем по этому отрезку и ищем первый подходящий сплоченный.

Значит мы для каждого фиксированного индекса i нашли минимальный j' такой что $[j', i]$ - сплочен. Минимальность j' дает максимальность суммы отрезка среди сплоченных отрезков с фиксированной правой границей i. Следовательно ответ можно обновлять только одним отрезком для каждого фиксированного i. Находя максимум среди них мы найдем ответ на задачу.

1.3 Затраты времени

- Чтение и вывод – $O(n)$
- Сортировка $QuickSort-$ по времени в среднем $O(nlogn)$
- Два указателя– по времени $O(n)$
- Итого – $O(nlogn)$

1.4 Затраты памяти

- Сортировка $QuickSort$ – $O(logn)$
- Структура с игроками $O(n)$ памяти
- Два указателя – $O(1)$
- Итого – $O(n)$