Тагове	**На пълното решение**	**На подзадачите**
	Грийди / Ad Hoc Сортировки	Пълно изчерпване

Анализ

Задачата не може да се определи само като грийди или като ад хок, но все пак смятам, че е грийди. Тя е втората ми задача с грийди за D група за годината. Първата беше на есенния СТИ, където резултатите не бяха от най-зелените. Надявам се, задачата да е по-добре ☺.

В задачата се търси оптимално разпределение на числа в редица по двойки, така че двойката с минимален сбор да има възможно най-малка разлика. Препоръчвам като четете анализа и спомена, че нещо може да се докаже, първо да се опитате сами да го докажете. Ако пък не можете, спокойно прочетете моето доказателство.

Решение за 20 точки

За всяка една пермутация на числата от 1 до N се разглежда дали спазва изискванията от условието. Намира се тази с минимална разлика на двойката с най-малка сума.

Постигната сложност: $O(N! \times N)$.

Имплементация: `First_20p.cpp`

Решение за 50 точки

Предвиденото решение разглежда всяка възможна двойка числа в редицата и проверява дали има пермутация, в която двойката е с минимална сума. Нека сме сортирали редицата и си представим, че сме отбелязали двойката, която проверяваме, с червено, а пък останалите числа със синьо. Нека сумата на двойката числа, отбелязани в червено, е s. Тогава искаме да намерим такова разпределение на сините числа по двойки, така че всяка двойка от тях да има сбор $\geq s$. Може да се забележи, че винаги е оптимално да групираме сините числа по следния начин: най-малкото синьо с най-голямото синьо, 2-рото най-малко синьо число в редицата с 2-рото най-голямо синьо, 3-тото най-малко с 3-тото най-голямо и т.н. Например за редицата 1 2 6 8 11 12 16 18 е оптимално числата да се групират по следния начин: (1,18); (2,16); (8,11). Защо? Да видим какво се случва, ако допуснем противното. Нека за четири числа $a \leq b \leq c \leq d$ да има решение като ги групираме (a, c) и (b, d), но няма решение като ги групираме (a, d) и (b, c). За да има решение при (a, c) и (b, d), то $a + c \geq s$ и $b + d \geq s$. Тъй като $b \geq a$, то и $b + c \geq a + c \geq s$. Тъй като $d \geq c$, то и $a + d \geq a + c \geq s$. Следователно, при $a + c \geq s$ и $b + d \geq s$, винаги $a + d \geq s$ и $b + c \geq s$, което означава, че има решение и за групиране (a, d) и (b, c). Противоречие. Сортираме числата, разглеждаме всяка двойка числа и обхождаме редицата, като проверяваме дали има решение.

Постигната сложност: $O(N^3)$.

Имплементация: `Second_50p.cpp`

Решение за 75 точки

Решението за 75 точки се основава върху идеята на гореописаното решение, но я оптимизира. Нека сме проверили дали има решение за червена двойка (i, j) и искаме да проверим за $(i, j+1)$. Нека сме групирали i с p_i. Въпросът е как се променя групирането. Всъщност, само две двойки се променят: от (i, j) и $(j+1, p_{j+1})$ се променят на $(i, j+1)$ и (j, p_{j+1}). Причината за това е, че ако числото на $j+1$ позиция в сортировката е било k-тото по големина, то след като $j+1$ стане червено, а j стане синьо, то j става k-тото по големина. Добре, как може да използваме това наблюдение? На нас ни трябва само да поддържаме най-малкия сбор на някоя синя двойка. Това може да стане със STL-ския multiset. Не очаквам да срещна решение точно за 75 точки, поради трудността му за писане за шестокласник, но съм винаги готов да видя някой ентусиаст ☺.

Постигната сложност: $O(N^2 log_2 N)$.

Имплементация: `Third_75p.cpp`

Решение за 100 точки

Нека забравим за идеята за 75 точки. За решението за 100 точки трябва да се забележи още едно нещо – винаги е оптимално да оцветим в червено две съседни числа в сортировката. Защо? Пак ще допуснем противното. Нека за 3 числа от редицата $a \leq b \leq c$ е възможно решение с червена двойка (a, c), но не е възможно такова за (a, b). Разбира се, по-оптимално е a да се групира с b, защото $b - a \leq c - a$. Нека в решението за (a, c) сме групирали b с x. За да има решение за (a, c), то сборът на числата в двойките е $\geq a + c$, включително $b + x \geq a + c$. Всъщност, ако се опитаме да намерим решение за (a, b) и групираме всички числа по същия начин, като c го групираме с x, то:

- Всяка двойка има сбор $\geq a + c \geq a + b$
- Двойката $c + x \geq b + x \geq a + c \geq a + b$

Следователно, когато има решение за (a, c), има и решение за (a, b). Противоречие.

Сортираме числата и прилагаме групирането от решението за 50 точки. Ако например, искаме да проверим дали може 4-тото и 5-тото число да образуват червена двойка, то те трябва да имат сбор $\leq \min(a_1 + a_N, a_2 + a_{N-1}, a_3 + a_{N-2})$.

Постигната сложност: $O(N\ log_2 N)$.

Имплементация: `duets_100p.cpp`

Автор: Борис Михов