**Анализ**

**Подзадача № 1**

В тази подзадача са тестовите примери. Тя е за обратна връзка от системата.

**Подзадача № 2**

Трябва да се започне от някъде, а именно от проверка за това дали един скобов низ е правилен. Един вариант е да се направи пълно изчерпване, следващо дефиницията от условието. Tо няма да е много приятно за писане от един шестокласник, а и ще е тромаво за изпълнение от един компютър (в най-добрия случай би работило достатъчно бързо за $N\leq 20$). Другият вариант е да се използва стандартната проверка за правилен скобов низ от един и същи вид скоби. Нека в един скобов низ заменим всяка отваряща скоба с $1$, а всяка затваряща скоба с $-$1. Трябва да важат следните две условия, за да е правилен:

* Сборът на всички числа е $0$. (броят на отварящите скоби е равен на броя на затварящите)
* Минималната префиксна сума е точно равна на $0$. (магия)

Тогава може лесно да се напише решение, обхождащо всеки един подниз и проверяващо дали е правилен.

Постигната сложност: $О(N^{3})$

Имплементация: brackets\_35p.cpp

**Подзадача № 3**

Решението на третата подзадача е дребна оптимизация на горното решение. Нека проверим дали е правилен всеки подниз с начало $l$. Може да се забележи, че поднизът $[l,r]$ ще има същите префикси като подниза $[l,r+1]$, изключвайки префикса, завършващ на $(r+1)$-вия елемент. Така ще трябва само да проверим за $(r+1)$-вия префикс, защото предишните ще са проверени.

Постигната сложност: $О(N^{2})$

Имплементация: brackets\_55p.cpp

**Подзадача № 4**

Нека в оригиналния низ заменим скобите с числа. Нека $p\_{i}$ е префиксната сума, завършваща на $i$-тия елемент, като $p\_{0}=0$. Тогава задачата бързо се свежда до това да преброим двойките $(l,r)$ ($1\leq l\leq r\leq N$), за които са в сила условията:

* $p\_{l-1}=p\_{r}$
* Няма $i$, за което $(l\leq i<r)$ и $p\_{i}<p\_{l-1}$

Ако горното условие важи за някоя двойка $(l,r)$, то поднизът $[l,r]$ е образува правилен скобов израз. Защо? Нека, като сме направили описаната проверка за някой подниз, да сме получили префиксни суми $a\_{1},a\_{2},…,a\_{r-l+1}$, където $a\_{i}$ е префиксната сума до $i$-тия символ на подниза. Тогава може да се забележи, че $a\_{1}=p\_{l}-p\_{l-1}$, $a\_{2}=p\_{l+1}-p\_{l-1}$, …, $a\_{r-l+1}=p\_{r}-p\_{l-1}$. За да бъде поднизът правилен скобов израз, то трябва $a\_{r-l+1}=0$ и всички $a\_{1},a\_{2},…,a\_{r-l+1}$ да са $\geq 0$. Горните две условия проверяват това.

Нека за момент игнорираме второто условие. Как да преброим двойките $(l,r)$ за които $p\_{l-1}=p\_{r}$? Тук има множество подходи, но този, който ще ни бъде най-полезен, е да съобразим, че числата са достатъчно малки, за да поддържаме counting. Въпросът е как да модифицираме counting-а, така че да реши нашата задача.

Нека в един масив cnt[value] поддържаме броя леви краища $l$ на поднизове, за които $p\_{l}=value$ и няма по-малък префикс след тях. Ние ще обхождаме възможните десни краища $r$ отляво-надясно, като ще променяме cnt спрямо $r$-тия префикс. Тогава има два случая:

* Текущата скоба е $($. Няма как скобов подниз, завършващ на $r$, да е правилен. Тогава текущият префикс става възможен ляв край за следващите десни краища, заради това увеличаваме $cnt[p\_{r}]$ с $1$. Тъй като префиксната сума се увеличава, то няма леви краища, за които $p\_{r}$ е първата по-малка следваща префиксна сума.
* Текущата скоба e $)$. Тогава за всеки ляв край с префиксна сума, равна на $p\_{r}+1$ се появява по-малък следващ префикс. Заради това трябва да занулим $cnt\left[p\_{r}+1\right]$. Всички леви краища с префиксна сума $>p\_{r}+1$ със сигурност са по-големи от $p\_{r-1}$, заради това те вече са занулени. Към отговора добавяме $cnt[p\_{r}]$.

И остана да отпечатаме верния отговор ☺.

Постигната сложност: $О(N)$

Имплементация: brackets\_100p.cpp

*Автор: Борис Михов*