**АНАЛИЗ**

**НА РЕШЕНИЕТО НА ЗАДАЧА**

**СКОБИ**

Задачата може да се реши с непосредствена проверка за всеки интервал [Li; Ri], но при по-малки N и T, или динамично - но пак с квадратна сложност.

Идеята за пълното решение е следната:

Нека редицата е ()(()(())))())

Понеже редицата се индексира от 1, за по-удобно поставяме лява скоба на позиция 0 и дясна скоба след последната.

Започваме от позиция 0 да увеличаваме брояч br с 1 ако срещнем лява скоба и го намаляваме с 1 ако срещнем дясна скоба. В масив A[i] записваме стойността на този брояч на позиция i.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| Редица | ( | ( | ) | ( | ( | ) | ( | ( | ) | ) | ) | ) | ( | ) | ) | ) |
| A[] | 1 | 2 | 1 | 2 | 3 | 2 | 3 | 4 | 3 | 2 | 1 | 0 | 1 | 0 | -1 | -2 |

Чрез масива A[] може веднага да кажем дали в интервала [L;R] има еднакъв брой леви и десни скоби. Ако на позиция L-1 имаме А[L-1]=K и в интервала има по равен брой Х различни скоби, то A[R] = K + X – X = К. Тогава изискването **A[R] = A[ L – 1]**е достатъчно, за да сме сигурни, че броят на скобите вътре е равен.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **L – 1** | **L** |  |  |  | **R** |
| Редица |  | X леви и X десни скоби | | | | |
| A[] | K |  |  |  |  | K+X – X |

Второто условие за [L;R] да е правилна последователност е следното: *да няма позиция в интервала, при която броят на десните скоби да става по-голям от броя на левите*. Например в този случай ()())() до позиция 5 имаме 2 леви и 3 десни скоби и няма за какво да проверяваме по-нататък – явно редицата не е правилна.

В общия случай, ако за първи път на позиция Р (L ≤ P ≤ R) десните стават повече от левите скоби, то е явно, че A[P] = A[K – 1] - 1. Заключението е, че позиция P е първата вдясно от позиция L – 1, в която A[P] < A[L – 1].

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **L – 1** | **L** |  |  | **P - 1** | **P** |  | **R** |
| Редица |  | X леви и X десни скоби | | | | ) |  |  |
| A[] | K |  |  |  | K + X - Х | K – 1 |  |  |

Има линеен алгоритъм, който намира за дадено число X кое /или къде/ е първото по-малко/по-голямо число вдясно/вляво от X. Резултатите се записват в масива B[].

Тогава, ако B[A[L – 1]] е в интервала [L;R], той не е правилен – трябва да е изпълнено B[A[L – 1]] > R.

Накратко, двете условия за интервала [L;R] да е правилен, са:

1. A[R] = A[ L – 1]
2. B[A[L – 1]] > R

Пример за [6;11]:

A[11] – A[5] = 0 – 2 = -2 не е изпълнено (1).

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| Редица | ( | ( | ) | ( | ( | ) | ( | ( | ) | ) | ) | ) | ( | ) | ) | ) |
| A[] | 1 | 2 | 1 | 2 | 3 | 2 | 3 | 4 | 3 | 2 | 1 | 0 | 1 | 0 | -1 | -2 |
| B[] | 11 | 2 | 11 | 10 | 5 | 10 | 9 | 8 | 9 | 10 | 11 | 15 | 13 | 15 | 15 |  |

Пример за [1;12]:

A[12] – A[0] = 1 – 1= 0 изпълнено (1).

B[A[0]] = B[1] = 11, R = 12 не е изпълнено (2), защото 11 < 12

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| Редица | ( | ( | ) | ( | ( | ) | ( | ( | ) | ) | ) | **)** | ( | ) | ) | ) |
| A[] | 1 | 2 | 1 | 2 | 3 | 2 | 3 | 4 | 3 | 2 | 1 | 0 | 1 | 0 | -1 | -2 |
| B[] | 11 | 2 | 11 | 10 | 5 | 10 | 9 | 8 | 9 | 10 | 11 | 15 | 13 | 15 | 15 |  |

Задачата е със сложност O(N+T). Може да се използва и двоично дърво, тогава сложността ще е O(N+TlogN).

*Автор, анализ, решение и тестове:*

*Павел Петров*