

Zadanie: ROB

Robot



Warsztaty ILO, grupa olimpijska, dzień 3. Dostępna pamięć: 128 MB.

Rozwiązanie wzorcowe $O(|s|^2)$

Pierwsza obserwacja jest taka, że możemy każdy wymiar rozważyć niezależnie – wprawdzie nie wiemy, w którą stronę skręca robot, ale wiemy, czy porusza się poziomo, czy pionowo. Zatem dla każdego ciągu liter F wiemy, czy robot poruszał się poziomo czy pionowo (będzie to na przemian, począwszy od pionowego). Każde T zmienia kierunek z pionowego na poziomy i odwrotnie.

Rozważmy zatem dla uproszczenia tylko poziome ruchy i pytamy się, czy da się dla każdego ciągu F wybrać, czy robot poruszał się wtedy w prawo czy w lewo tak, że ostatecznie skończył w punkcie x . Nietrudno zauważyć, że jest to zagadnienie standardowego programowania dynamicznego. Mamy ciąg liczb i dla każdej z nich chcemy wybrać jej znak (z plusem lub z minusem), aby sumowały się do x . Plus oznacza pójście w prawo, a minus w lewo. Takich liczb mamy do $\frac{|s|}{2}$, a sumują się one co najwyżej do $|s|$. Zatem będziemy przechowywać tablicę booli, która będzie trzymać, które sumy jesteśmy w stanie osiągnąć. Początkowo wartość 1 jest jedynie w punkcie 0. Gdy dodajemy jakąś liczbę, aktualizujemy tę tablicę. Przykładowo jeśli mogliśmy otrzymać sumę a , to będziemy mogli również otrzymać sumę $a + w_i$ oraz $a - w_i$, gdzie w_i jest dodawaną liczbą. Na koniec sprawdzamy, czy da się otrzymać sumę x . To rozwiązanie ma złożoność kwadratową $O(|s|^2)$. To rozwiązanie można przyspieszyć korzystając z bitseta do obliczania tablicy, ale nie było to konieczne do otrzymania maksymalnej liczby punktów.

Analogicznie rozwiązujemy problem dla ruchów pionowych. Należy jednak pamiętać o tym, że pierwszy ruch zawsze jest w górę, jedynie dla pozostałych możemy sobie wybrać kierunek góra/dół.