



Jasio znalazł się w krainie kamieni. Sercem tej krainy jest idealnie równy rząd N kamieni. Każde dwa sąsiednie kamienie odległe są o jeden bajtometr.

Jasio postanowił skakać po tych kamieniach. Żeby nie było za łatwo nie będzie przeskakiwał po kolei, zgodnie z ustawieniem kamieni w rzędzie, lecz opracował skomplikowany układ skoków – dokładną kolejność kamieni, na które będzie kolejno wskakiwał. Aby nie było nudno ani smutno – na każdy kamień wskoczy dokładnie raz.

Z układem Jasia jest pewien problem. Niektóre kamienie pomiędzy którymi Jasio chce skakać mogą być bardzo odległe. Jasio zastanawia się jaki najdłuższy spójny kawałek jego układu będzie w stanie wykonać w miarę treningu coraz dłuższych skoków. Pomóż mu!

Napisz program, który: wczyta układ Jasia, wyznaczy dla każdej długości skoku Jasia, jaki jest najdłuższy kawałek zaplanowanej sekwencji jego skoków, który będzie w stanie wykonać przy tej długości skoku i wypisze wyniki na standardowe wyjście.

WEJŚCIE

W pierwszym wierszu wejścia znajduje się jedna liczba naturalna N , określająca liczbę kamieni (a zarazem długość sekwencji Jasia). W drugim (ostatnim) wierszu wejścia znajduje się ciąg N liczb naturalnych A_i . i -ta liczba w ciągu określa moment, w którym Jasio chce wskoczyć na i -ty kamień.

Kamienie numerujemy (zgodnie z kolejnością ich występowania w rzędzie) kolejnymi liczbami naturalnymi od 1 do N włącznie.

WYJŚCIE

Twój program powinien wypisać na wyjście $N - 1$ wierszy. W i -tym wierszu powinna się znaleźć odpowiedź – maksymalna liczba kolejnych skoków z opracowanego ciągu skoków Jasia, którą można wykonać przy ograniczeniu maksymalnej długości pojedynczego skoku Jasia do i bajtometrów.

OGRANICZENIA

$1 \leq N \leq 500\,000$.

PRZYKŁAD

Wejście

8
7 4 1 3 2 5 6 8

Wyjście

1
3
3
5
5
6
7

Przykładowo, dla maksymalnej długości skoku 2, Jasio może skoczyć między pierwszymi czterema kamieniami jego sekwencji (na pozycjach: trzeciej, piątej, czwartej oraz drugiej).