

Akcelerator

Kurs programowania i algorytmiki OI: kurs.oi.edu.pl

Kod zadania: **akc**
Limit czasu: **2 s**
Limit pamięci: **256 MB**



Fyzyk-stażysta Bajtazar śledzi działanie Wielkiego Bajtockiego Akceleratora Cząstek. W akceleratorze porusza się duża liczba cząstek o różnych prędkościach (dodatnich albo ujemnych, w zależności od kierunku ruchu). Zadaniem jest mierzenie tych właśnie prędkości.

Bajtazar wykrył n cząstek i zmierzył ich prędkości. Z braku lepszych zajęć ustawił wszystkie wyniki pomiarów w kolejności niemalejącej. Opracowanie wyników wymaga jednak odpowiedzi na kilka pytań postaci „dla zadanej prędkości, ile jest cząstek, które poruszały się z tą właśnie prędkością?”

Pomóż mu znaleźć odpowiedzi i zakończyć staż z pozytywną oceną!

Wejście

W pierwszym wierszu wejścia znajduje się liczba całkowita n ($1 \leq n \leq 10^5$) oznaczająca liczbę cząstek. W drugim wierszu znajduje się n liczb całkowitych o wartości bezwzględnej nie przekraczającej 10^9 , oddzielonych spacjami — są to kolejne prędkości cząstek, uporządkowane niemalejąco. W trzecim wierszu znajduje się liczba całkowita q ($1 \leq q \leq 10^6$) oznaczająca liczbę zapytań, które ciekawią Bajtazara. Kolejnych q wierszy zawiera po jednej liczbie całkowitej, której wartość bezwzględna jest nie większa niż 10^9 — są to prędkości, o które pyta Bajtazar.

Wyjście

Na wyjście wypisz dokładnie q wierszy. Wiersze te powinny zawierać odpowiedzi na kolejne pytania — odpowiedzią jest ilość wystąpień podanej liczby wśród odczytów.

Wskazówki

Nie próbuj przy każdym zapytaniu przejeżdżać pętlą przez wszystkie odczyty, jest to bowiem zbyt czasochłonne i prawie na pewno spowoduje komunikat *Przekroczenie limitu czasu*. Dobrym pomysłem na pierwszy krok jest znalezienie szukanej liczby w tablicy za pomocą wyszukiwania binarnego. Nie powinieneś jednak potem iterować się po elementach tablicy, szukając wszystkich wystąpień tej liczby — znowu, to również spowoduje, że Twój program będzie działał zbyt długo. Pamiętaj, że w algorytmice zwykle zakłada się, że Twój program dostanie nieprzyjemne dane wejściowe — postaraliśmy się zatem o możliwie trudne testy!

Wejście dla testu akc0:

```
5
1 1 2 4 5
3
1
2
3
```

Wyjście dla testu akc0:

```
2
1
0
```