

Próg kwalifikacyjny

XIV OIJ, zawody I stopnia, tura ukryta
14 października 2019 – 13 stycznia 2020

Kod zadania: **kwa**
Limit czasu: **5 s**
Limit pamięci: **256 MB**



Bajtazar startuje w Bajtockiej Olimpiadzie Informatycznej Juniorów (BOIJ). Zasady tej olimpiady nieco różnią się od OIJ: do rozwiązania na pierwszym etapie jest pewna liczba zadań, a za każde z nich można uzyskać ustaloną liczbę punktów (być może różną dla różnych zadań), która jest przyznawana tylko, jeśli zadanie zostanie w pełni poprawnie rozwiązane. W innym przypadku rozwiązanie otrzymuje zawsze zero punktów.

Bajtazar jest dość leniwy, dlatego zamiast zabrać się do rozwiązywania zadań, zastanawia się ile zadań będzie wystarczające, żeby dostać się do drugiego etapu. Co roku, po zawodach, Komitet Główny BOIJ publikuje próg kwalifikacyjny do drugiego etapu zawodów. Każdy zawodnik, który będzie miał przynajmniej tyle punktów, ile wynosi próg, zostanie zakwalifikowany. Niestety, sytuację utrudnia fakt, że Bajtazar nie zna progu w trakcie zawodów. Każdej nocy jednak ma sen, w którym dowiadyuje się ile wynosi wartość progu – co rano zastanawia się wtedy, ile zadań wystarczyłoby w takim wypadku zrobić. Napisałby program odpowiadający mu na to pytanie, jednak, no właśnie, jest trochę leniwy. Dlatego poprosił Cię o pomoc.

Napisz program, który wczyta wartości punktowe poszczególnych zadań oraz potencjalne progi kwalifikacyjne, dla każdego progu wyznaczy minimalną liczbę rozwiązanych zadań niezbędnych do przejścia do drugiego etapu i wypisze wyniki na standardowe wyjście.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba naturalna N ($1 \leq N \leq 200000$), określająca liczbę zadań na pierwszym etapie zawodów. W drugim wierszu wejścia znajduje się ciąg N liczb naturalnych V_i ($1 \leq V_i \leq 10^9$), pooddzielanych pojedynczymi odstępami – i -ta liczba określa liczbę punktów za poprawne rozwiązanie i -tego zadania.

W trzecim wierszu wejścia znajduje się jedna liczba naturalna Q ($1 \leq Q \leq 200000$), określająca liczbę snów Bajtazara. W kolejnych Q wierszach znajduje się opis tych snów, po jednym w wierszu. Opis każdego snu składa się z jednej liczby naturalnej P_i , $1 \leq P_i \leq V_1 + V_2 + \dots + V_N$, określającej wyśniony próg do drugiego etapu zawodów podczas i -tego snu Bajtazara.

Wyjście

Twój program powinien wypisać na wyjście dokładnie Q wierszy: po jednym dla każdego snu Bajtazara. W i -tym wierszu powinna się znaleźć jedna liczba naturalna – minimalna liczba zadań do rozwiązania, wystarczająca do kwalifikacji do drugiego etapu przy założeniu wyśnionego progu z i -tego snu.

Przykład

Wejście dla testu kwa0a:

```
6
3 2 1 1 5 1
3
3
13
9
```

Wyjście dla testu kwa0a:

```
1
6
3
```

Wyjaśnienie do przykładu: Jeżeli próg wyniósłby 3 punkty, to wystarczy rozwiązać jedno zadanie – to warto 3 punkty, bądź to warto 5 punktów. Jeżeli natomiast próg wyniósłby 13 punktów, to wymagane byłoby rozwiązanie wszystkich sześciu zadań. Z kolei jeżeli próg wyniósłby 9 punktów, to wystarczyłoby rozwiązać trzy zadania, na przykład te warto 2, 3 oraz 5 punktów, co dawałoby 10 punktów.

Pozostałe testy przykładowe

- test kwa0b: $N = 4$, $Q = 10$, $V_i = i$, $P_i = i$
- test kwa0c: $N = 5$, $Q = 6$, $V_i = 2^i$, $P_i = 10 \cdot i$
- test kwa0d: $N = 100$, $Q = 50$, $V_i = i$, $P_i = i^2$
- test kwa0e: $N = 200000$, $Q = 200000$, punktacja zadań to naprzemienny ciąg $2, 3, 2, 3, \dots$, wyznione progi są losowe

Ocenianie

Poniższa tabela opisuje dodatkowe warunki, które spełniają pewne grupy testów oraz liczbę punktów, którą można otrzymać za rozwiązanie jedynie testów spełniające te warunki.

Dodatkowe ograniczenia	Liczba punktów
Wszystkie zadania są warte 1, 2 lub 3 punkty.	30
$Q \leq 10$	35