



## Maximale Lücke

Task ID: maxgap	Time limit: 0.6 sec	Memory limit: 256 MB
-----------------	---------------------	----------------------

### Eingabe

Die erste Zeile enthält  $N \leq 10^6$ . Die zweite Zeile enthält  $a_1, \dots, a_N$ , wobei  $0 \leq a_i \leq 10^9$  für alle  $1 \leq i \leq N$ . Beachte, dass diese Zahlen  $a_i$  nicht sortiert sein müssen.

### Limits

Die Eingabe besteht aus 4 Testgruppen, jede gibt 25 Punkte.

- Für 25 Punkte gilt  $N \leq 10^3$  (erlaubt  $\mathcal{O}(N^3)$  Ansätze).
- Für 50 Punkte gilt  $N \leq 10^4$  (erlaubt  $\mathcal{O}(N^2)$  Ansätze).
- Für 75 Punkte gilt  $N \leq 10^5$  (erlaubt  $\mathcal{O}(N \cdot \log N)$  Ansätze).
- Für 100 Punkte gilt  $N \leq 10^6$  (erlaubt  $\mathcal{O}(N)$  Ansätze).

### Ausgabe

Gib die Grösse der maximalen Lücke aus. Eine Lücke ist die Differenz zweier Zahlen aus der Eingabe, sodass keine andere Zahl aus der Eingabe dazwischen liegt. Also  $\max_{1 \leq i, j \leq N} (a_j - a_i)$ , sodass gilt

- $a_i \leq a_j$
- $\forall k : a_k \leq a_i \vee a_k \geq a_j$

### Beispiele

input	output
7 20 12 3 5 16 2 24	7

### Hinweise

- Der Grader für diese Zusatzaufgaben akzeptiert Einsendungen in Java, C, C++, Python und Pascal.
- Für Einsendungen in Java muss die Klasse *submission* heissen, vgl. die Templates auf der Webseite.
- Da die Eingabe sehr gross werden kann, beachtete folgende Hinweise:
  - Für Java: verwende `BufferedReader` statt `Scanner` (siehe Template).
  - Für C/C++: Benutze entweder C/C++ `printf/scanf` oder verbessere die Performance von C++ streams indem du `std::ios_base::sync_with_stdio(false);` am Anfang von *main* inkludierst.