

Taak 6 – Rijbewijs (driverlicence) – (100 ptn) (A: 11 ptn, B: 28 ptn, C: 29 ptn, D: 32 ptn)

Het rijexamen wordt hervormd om beter aan te sluiten bij de veranderende samenleving: vanaf nu gaat men meten hoe goed een bestuurder op zijn rijvak kan blijven terwijl die zijn/haar smartphone programmeert. Er zijn net nieuwe sensoren op de markt die toelaten om erg precies te meten hoe goed men op koers blijft.

Voor het begin van elk examen wordt een tolerantielimiet T gekozen. De bestuurder moet zo lang mogelijk blijven rijden zonder meer dan T van het traject af te wijken.

Taak

Je moet een algoritme schrijven dat, gegeven een getal T en een reeks metingen, de lengte vindt van de langste rij opeenvolgende metingen die binnen de tolerantielimiet T liggen (d.w.z., waarbij het verschil tussen het minimum en het maximum van de reeks kleiner dan of gelijk is aan T).

Limieten en beperkingen

- $1 \leq N \leq N_{MAX}$, het aantal metingen dat werd gedaan.
- $0 \leq T \leq T_{MAX}$, de maximaal getolereerde limiet.
- $1 \leq m_i \leq T_{MAX}$, een meting.

	N_{MAX}	T_{MAX}
Subtaak A	200	20 000
Subtaak B	10 000	1 000 000 000
Subtaak C	300 000	1 000 000 000
Subtaak D	3 000 000	2 000 000 000

Maximale uitvoeringsduur: **2 seconden**. Geheugenlimiet: **512 MB**.

Input

Jouw programma krijgt input in het volgende formaat:

- De eerste lijn bevat T en N , gescheiden door een spatie.
- De volgende lijn bevat N metingen m_i , gescheiden door een spatie.
- De input eindigt met een nieuwe lijn.

Output

De output van jouw programma bevat één enkel getal, namelijk de lengte van de langste reeks metingen die binnen de tolerantielimiet T blijven. De output eindigt met een nieuwe lijn.

Voorbeeld

Voor de volgende input:

```
3 9
5 1 3 5 8 6 6 9 10
```

moet jouw programma teruggeven:

```
4
```

Uitleg: Er zijn twee langste secties die binnen de tolerantielimiet van 3 blijven, met lengte 4: 5, 8, 6, 6 en 8, 6, 6, 9.