

## Zadanie U Reaktor

Wygląda na to, że Bajtocjanom nie jest dane spać spokojnie (a przynajmniej nie przed końcem semestru) - ich kraj nękanym jest przez coraz to nowe i coraz poważniejsze problemy. Tym razem Bajtocja stanęła w obliczu kryzysu energetycznego. Wskutek konfliktu dyplomatycznego z sąsiadującym mocarstwem całkowicie wstrzymane zostały dostawy gazu, podstawowego surowca energetycznego. Z braku alternatywy władze postanowiły przyspieszyć uruchomienie eksperymentalnego prototypu supernowoczesnego ekologicznego źródła energii - reaktora neutronowego.

Pracami nad budową i uruchomieniem reaktora zajmuje się specjalne Centrum Energetyki Reaktora Neutronowego (CERN), zrzeszające najwybitniejszych bajtocjańskich naukowców. Jak to zazwyczaj z wybitnymi naukowcami bywa, zupełnie nie mają oni czasu ani ochoty do wykonywania niektórych czynności, na przykład tworzenia oprogramowania. CERN zwrócił się więc do Ciebie z prośbą o pomoc w napisaniu programu zajmującego się kontrolą reaktora.

Reaktor neutronowy ma kształt długiego tunelu składającego się z  $n$  połączonych szeregowo komór, ponumerowanych od 0 do  $n - 1$ . Z obu końców tunelu wystrzeliwane są naprzeciw siebie wiązki neutronów, które po zderzeniu w jednej z komór powodują błyskawiczną reakcję jądrową. Wydzielona w jej wyniku energia jest następnie absorbowana przez zgromadzone w ścianach komory akumulatory, skąd może zostać odprowadzona dalej. Akumulatory wyposażone są w odpowiednie czujniki bezpieczeństwa, które sygnalizują ich przepełnienie. Czujniki te nie działają dobrze w wysokich temperaturach. Równocześnie, długotrwałe przegrzanie i przepełnienie akumulatora grozi wybuchem. Sytuacja ta wymaga w rejonach mocno nasłonecznionych przeprowadzania częstej kontroli pojemności.

Twoim zadaniem będzie napisanie programu monitorującego stan tych właśnie akumulatorów. Musi on obsługiwać następujące komendy:

- **INS  $i$   $v$**  – w  $i$ -tej komorze nastąpiła reakcja w wyniku której wydzielilo się  $v$  ( $v$  jest liczbą całkowitą z przedziału  $[1, 1000]$ ) jednostek energii. Jeżeli w tej komorze była już zgromadzona jakaś energia, wartości te dodają się.
- **DEL  $i$**  – odprowadzono energię zgromadzoną w  $i$ -tej komorze. Program powinien wypisać liczbę odprowadzonych jednostek. Jeżeli odprowadzono energię z „pustej” komory, należy wypisać **ERROR**.
- **SUM  $a$   $b$**  – należy wypisać łączną liczbę jednostek energii zgromadzonych w spójnym fragmencie reaktora obejmującym komory od  $a$ -tej do  $b$ -tej włącznie ( $a \leq b$ ).
- **AVE  $a$   $b$**  – należy wypisać średnią liczbę jednostek energii zgromadzonych w **aktywnych** komorach od  $a$ -tej do  $b$ -tej włącznie ( $a \leq b$ ). Komora jest aktywna, jeśli jest w niej zgromadzona jakaś energia. Otrzymaną średnią należy zaokrąglić do najbliższej liczby całkowitej (Liczba 0.5 zaokrąglana jest do góry). Jeżeli w podanym przedziale nie ma aktywnych komór, należy wypisać **ERROR**.

- **NXT**  $i$  – należy wypisać numer pierwszej aktywnej komory następującej po  $i$ -tej wraz w informacją o zgromadzonej w niej energii. Liczby należy wypisać w jednej linii, oddzielone spacją. Jeżeli nie ma takiej komory (wszystkie spośród  $i + 1, \dots, n - 1$  są nieaktywne) należy wypisać **ERROR**.
- **MAX**  $a$   $b$  – należy wypisać liczbę jednostek energii zgromadzoną w najbardziej pełnej aktywnej komorze znajdującej się w spójnym fragmencie reaktora obejmującym komory od  $a$ -tej do  $b$ -tej włącznie ( $a \leq b$ ). Jeżeli w przedziale nie ma aktywnej komory należy wypisać 0.

Każda z powyższych operacji powinna mieć pesymistyczny czas wykonania logarytmiczny względem liczby komór. W rozwiązywaniu zadania należy zastosować drzewo licznikowe.

## Wejście

Pierwsza linia wejścia zawiera liczbę całkowitą  $z$  ( $1 \leq z \leq 2 \cdot 10^9$ ) – liczbę zestawów danych, których opisy występują kolejno po sobie. Opis jednego zestawu jest następujący:

Pierwsza linia zestawu zawiera dwie liczby całkowite dodatnie  $n, k$  ( $n, k \leq 10^7$ ) oznaczające odpowiednio liczbę komór w reaktorze oraz liczbę komend, które musi wykonać program. Każda z kolejnych  $k$  linii zawiera jedną komendę. Możesz założyć, że liczby, którymi kiedykolwiek będzie musiał operować program są mniejsze niż  $2 \cdot 10^9$ .

## Wyjście

Dla każdej komendy z wyjątkiem **INS** wypisz w osobnej linii odpowiedź zgodną z wymaganą przez CERN specyfikacją.

**Wersja U1** - nie obsługuje poleceń: **AVE**, **NXT**, **MAX**, wersja za 0.5 pkt.

**Wersja U2\*** - obsługuje wszystkie polecenia, wersja za dodatkowe 0.5 pkt.

**Dostępna pamięć: 512MB**

## Przykład

Dla danych wejściowych:

```
1
8 20
INS 0 3
INS 4 2
INS 7 6
SUM 0 7
SUM 0 5
INS 5 3
INS 4 4
SUM 3 7
AVE 0 7
MAX 0 6
NXT 3
DEL 4
SUM 3 7
MAX 2 6
NXT 0
NXT 3
NXT 5
NXT 7
MAX 1 3
AVE 1 3
```

Poprawną odpowiedzią jest:

```
11
5
15
5
6
4 6
6
9
3
5 3
5 3
7 6
ERROR
0
ERROR
```