Bajtex S.A.

Limit pamięci: 40 MB

Zbliża sie koniec roku, więc poszczególne zespoły spółki Bajtex S.A. powinny podsumować roczne osiągnięcia i ustalić strategię na przyszły rok. Spółka ma coraz więcej zespołów, a pomieszczeń w biurowcu wciąż tyle samo... Postanowiono więc na wyższych szczeblach, aby meetingi niektórych zespołów połączyć i przeprowadzić we wspólnych pomieszczeniach. Jako etatowy deweloper Bajtexu, zostałeś wyznaczony do napisania systemu umożliwiającego sprawne zaplanowanie meetingów. Napisany przez Ciebie program, zgodnie ze specyfikacją, obsługuje dwa typy operacji:

- (**P** x y) Połącz spotkania - w nowym planie meetingi, w których będą uczestniczyć zespoły x oraz y zostają połączone w jedno spotkanie (które siłą rzeczy odbędzie się więc jednym pomieszczeniu).
- (**C** k) Cofnij k **ostatnich** operacji połączenia spotkań (operacji typu **P**). Operacja ta może przydać się w przypadku drobnej pomyłki w procesie planowania.

Spełniłeś swoje zadanie i napisałeś system. W niecały tydzień! Gratulacje!! A teraz... już za chwilę sylwester i świętowanie z przyjaciółmi do rana! Oby tylko nie pilna sprawa w pracy...

Bitena (CEO): *O, Bajtosław! Twój system sprawuje się znakomicie! Nie masz chyba planów na sylwestra?*

Ty (Zwykły developer z nizin korporacyjnej hierarchii): *W sumie to [Nie zdążyłeś dokończyć zdania]*

Bitena (CEO): *Nie masz?! To wspaniale! Otóż...*

...i tak sobie rozmowa szła dalej

Streszczając, pojawił się pewien problem z optymalizacją spotkań... Otóż na bieżąco wychodzą na jaw wewnętrzne spory między zespołami. Management raz po raz dostaje telefon od kierownika któregoś z zespołów (nazwijmy go x), mówiący mniej więcej *"Śmieszne! Możecie sobie jedynie pomarzyć o tym, by NASZE spotkanie odbyło się w tym samym pomieszczeniu, co spotkanie tych nieudaczników z zespołu y..."*. Management doszedł do wniosku, iż wystarczy im możliwość realizacji następującego zapytania:

- (**Z** x y) *Ile wynosi minimalna wartość k taka, że po wykonaniu operacji "***C** k*", spotkania zespołów x oraz y odbyłyby się w różnych pomieszczeniach?* (Po uzyskaniu takiej informacji wiadomo będzie, czy warto dokonać cofnięć, czy też lepiej po prostu zignorować wymagania kierownika niesfornego zespołu)

Napisz nową wersję systemu przed sylwestrem, a może nawet zdążysz się trochę pobawić!

Wejście

W pierwszym wierszu dane są dwie liczby n ($2 \leq n \leq 10^6$) oraz m ($1 \leq m \leq 1.5 \cdot 10^6$) oznaczające odpowiednio liczbę zespołów oraz liczbę operacji, które należy zasymulować. Zespoły są numerowane od 1 do n. W każdym z kolejnych m wierszy znajduje się opis jednej operacji zgodny z formatem przedstawionym w opisie operacji. Gwarantowane jest, że każda operacja jest wykonalna. Dokładniej, gdy należy wykonać operację "**C** k", gwarantowane jest iż istnieje co najmniej k wykonanych i niecofniętych dotychczas operacji typu **P**. Ponadto nie wystąpi nigdy zapytanie w postaci "**Z** x x". Możliwa jest natomiast operacja "**P** x y" nawet wtedy, gdy zespoły x i y już mają zaplanowane wspólne spotkanie. Początkowo każdy zespół ma zaplanowane spotkanie w oddzielnym pokoju.

Wyjście

Należy wypisać wynik dla każdej operacji typu **Z**, w tej samej kolejności, w jakiej pojawiły się na wejściu.

UWAGA! Aby zminimalizować wpływ czasu wczytywania / wypisywania danych na uzyskany wynik, udostępniona została w zakładce *"pliki i testy"* bilbioteczka *"fastio.h"* wraz z przykładowym rozwiązaniem *"r4e.cpp"*, które z niej korzysta. Zachęcamy do wysłania tego rozwiązania w serwisie szkopuł. Otrzymuje ono jeden punkt. Biblioteczka udostępnia funkcje do szybkiego wczytywania danych. Sugerujemy użycie dostarczonych funkcji **zamiast** cin / cout. Nie jest to jednak formalny wymóg. Aby użyć biblioteczki, wystarczy umieścić ją w tym samym folderze co program, natomiast w programie użyć dyrektywy *"#include fastio.h"*. Dokładny opis funkcjonalności znajduje się w pliku.

Bajtex S.A.

Limit pamięci: 40 MB

Przykłady

Wejście dla testu `r4e0t0`:

```
5 15
Z 4 5
P 4 5
P 1 2
Z 4 5
P 1 4
P 3 4
P 1 4
Z 1 5
C 3
P 1 5
Z 2 5
P 4 5
Z 4 5
Z 2 4
P 4 4
```

Wyjście dla testu `r4e0t0`:

```
0
2
3
1
4
2
```

Wyjaśnienie: W momencie pierwszego zapytania, zespoły 4 oraz 5 mają zaplanowane spotkania w oddzielnych pokojach. Nie trzeba więc cofać żadnych operacji. W momencie drugiego zapytania, trzeba by cofnąć wszystkie operacje typu **P**, gdyż już w pierwszej z nich "**P** 4 5" spotkania zespołów 4 oraz 5 zostały połączone. W momencie trzeciego zapytania, aby meetingi zespołów 1 oraz 5 odbyły się w oddzielnych pomieszczeniach, konieczne okazałoby się usunięcie trzech ostatnich operacji (gdyż spotkania zespołów 1, 5 zostały połączone w jedno operacją "**P** 1 4"). Następna operacja "**C** 3" usuwa właśnie te operacje.

Ocenianie

Podzadanie	Ograniczenia	Punkty
1	Grupa zawiera dokładnie jeden test ($n = 2, m = 1$)	1
2	$n = 2, m \leq 100$	1
3	$n \leq 100, m \leq 200$	3
4	$n \leq 1000, m \leq 2000$	5
5	$n \leq 10^5, m \leq 2 \cdot 10^5$, nie występują cofnięcia	10
6	$n \leq 10^4, m \leq 2 \cdot 10^4$	15
7	$n \leq 10^5, m \leq 2 \cdot 10^5$	15
8	brak dodatkowych ograniczeń	50

Limit czasu: 1.5s (C++) / 90s (Python)
Limit pamięci dla Pythona wynosi **256 MB**
Wyższe limity wcale nie znaczą, że prościej się w nich zmieścić (;

- Test **r4e1ocen.in** jest taki sam jak test pierwszy z grupy pierwszej
- Test **r4e2ocen.in** $n = 2$, $m = 10$.
- Test **r4e3ocen.in** $n = 100$, $m = 200$.
- Test **r4e4ocen.in** $n = 1000$, $m = 2000$.
- Test **r4e5ocen.in** $n = 10^5$, $m = 2 \cdot 10^5$, nie występują cofnięcia.
- Test **r4e6ocen.in** $n = 10^4$, $m = 2 \cdot 10^4$.
- Test **r4e7ocen.in** $n = 10^5$, $m = 2 \cdot 10^5$.
- Test **r4e8ocen.in** $n = 10^6$, $m = 1.5 \cdot 10^6$.