Samolot

PREOI 2025 Kod zadania: **sam**
Dzień 3 – 27 stycznia 2025

Podzadanie 1 ($1 \leq n, m, q \leq 200$)

Dla każdego z q zapytań można przejść się po kolei po uruchamianych maszynach, których jest maksymalnie $n * m$. Niech ost_k będzie ostatnim dniem, w którym $k - ta$ maszyna została uruchomiona. Wtedy wynik dla rozpatrywanego zapytania i zwiekszamy o 1 jeśli różnica aktualnego dnia i ost_k jest większa od s_i. To rozwiązanie ma złożoność $O(nmq)$.

Obserwacja 1

Zauważmy, że dla każdego uruchomienia pewnej maszyny zwiększa ona wynik o 1 dla zapytań, dla których jest spełnione $s_i < D - ost_k$, gdzie D to dzień, w którym ta maszyna została uruchomiona. Uruchomienia tej maszyny tego dnia nia ma wpływu na pozostałe zapytania.

Podzadanie 2 ($1 \leq n, m \leq 2000$)

Korzystając z tej obserwacji można dla każdego dnia policzyć $\Delta D = D - ost_k$ przechodząc sie raz po wszystkich $O(n * m)$ dniach. Wtedy sortując zapytania i ΔD jesteśmy w stanie odpowiadać na zapytania.

Obserwacja 2

Zauważmy, że podczas jednego zadania działają kolejno maszyny z przedziału $[l_i; r_i]$ dniach, dlatego ich wartości ost_k po tym zadaniu będą miały wartości z pewnego przedziału.

Podzadanie 3 ($1 \leq m \leq 2000$)

Korzystając z obserwacji 2 nie musimy juz trzymać ost_k dla każdej z n maszyn tylko możemy trzymać te same informacje w postaci trójek (p, k, t), które oznaczają $ost_p = t$, $ost_{p+1} = t + 1$, ..., $ost_k = t + k - p$. Możemy wyznaczyć, które przedziały maszyn zachowują się tak samo przez cały przebieg budowy. Niech $A = \{1\} \cup \{l_i\} \cup \{r_i + 1\}$. Wtedy po posortowaniu tego zbioru każda para kolejnych elementów tworzy przedział maszyn, które zachowują się identycznie, tzn. należą do tej samej trójki przez cały przebieg budowy. Zatem żeby rozwiązać to podzadanie wystarczy zrobić to samo co w poprzednim, tylko zamiast n pojedynczych maszyn mamy teraz m przedziałów maszyn, które w obrębie jednego przedziału należą do tej samej trójki. Jako, że ΔD jest taka sama dla każdej maszyny z jednej trójki, to do wyniku należy dodawać $k - p + 1$, bo dla tyle jest maszyn o takim ΔD w rozpatrywanej trójce. Zamiast aktualizować ost_k trzeba analogicznie aktualizować wartości t w trójkach (p, k, t).

Podzadanie 4 ($l_i = 1$) oraz wzorcówka

W rozwiązaniu 3 podzadania ustaliliśmy trójki maszyn na początku rozwiązania, a potem już ich nie modyfikowaliśmy. Teraz będą takie modyfikacje. Będziemy symulować kolejne zadania, po każdym z nich aktualizując trójki. Gdy symulujemy zadanie i, interesują nas trójki których przedziały maszyn przecinają się z $[l_i; r_i]$. Po aktualizacji powstanie nowa trójka (l_i, r_i, D), gdzie D to number dnia i-tego zadania. Poprzednio istniejące trójki, które są całkowicie pokryte przez nową trójke zostaną usunięte, a trójki pokrywające się z prawej i lewej strony zostaną skrócone, tak żeby nie pokrywały nie z nowo powstałą trójką. W przypadku gdy istniała trójka, która pokrywa przedział $[l_i, r_i]$, zostanie ona podzielona na 2 częsci. Przy usuwaniu i skracaniu trójek należy dodawać nowe ΔD w odpowiednich ilościach, tak jak w poprzednich podzadaniach. Te wszystkie operacje można zaimplementować za pomocą seta, wtedy rozwiązanie ma złożoność $O(m * logm + q * logq)$. Podzadanie 4 jest prostrzą wersją tej implementacji, która nie wymaga rozpatrywania wszystkich przypadków. W implementacji tego podzadania wystarczy użyć stosu zamiast seta.