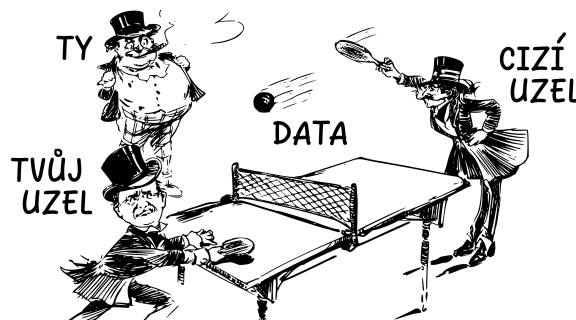


Úloha č. 2

Network Abusal Storage



Odpověz sfinze!

10 b

Tato úloha je vyhodnocována automaticky. Je potřeba, aby výstup programu přesně korespondoval se specifikací výstupu níže. Jak odevzdávat tento typ úloh se můžeš dočíst na webových stránkách FIKSu pod záložkou „Jak řešit FIKS“.

Hned, jak jsem zapojil poslední kabel, se z různých zařízení začaly linout různé zvuky. Různé frekvence různých zdrojů tvořily podivné přístrojové akordy. Pípání modemu se střídalo se zvukem CD mechaniky, která se opakovaně vysouvala a zasouvala neobsahující žádný disk. Občas zavádila o vypojený kabel, který posunula do rotujícího větráčku, aby začal mlátit o jednotlivé lopatky. Do cvakotu relátek se postupně přidávalo hučení pevných disků a pískot ventilátorů. Zařízení se spouštěla a vypínala, jako sekce orchestru nastupují a ustávají v symfonii. Jak skladba gradovala, uslyšel jsem otočení ventilu a zvuk tekoucí vody protékající zvláštním systémem hadiček. Z konců hadiček voda odkapávala na přehřáté komponenty, na kterých se se syčením měnila v páru. Asi vodní chlazení.

Pára mísící se s houstnoucími oblaky prachu postupně sytila zbytky vzduchu v místnosti. Světlo z jednotlivých zařízení postupně splývalo v jednu kontinuální zář. Rohy místnosti se však v mlze a prachu neztrácely. Byly vidět úplně stejně jako stěny. Udělal jsem pár kroků směrem k zařízením a následoval obvod místnosti. K mému překvapení jsem ale došel opět na původní místo, aniž bych narazil na roh.

To však již upoutal moji pozornost nový text na CRT monitoru. Tedy to jsem si alespoň myslel. Teď to spíše vypadalo, že obraz leží přímo v mlze, neboť CRT monitor jsem nemohl nahmatat.

„Máme problém, Frodo. Proces injekce již započal. Nyní to ale vypadá, že nám došla paměť. Podívej se po tvém uzlu a zkus nějakou najít.“

Uzlu? Asi myslí tuto místnost, pomyslel jsem si. V první krabici na zemi byla jen spousta síťových karet, v druhé se na dně válelo asi půl třetího tuctu vymačkaných kláves enter. Pět krabic směrem, kde bývaly dveře, jsem už prošel, když jsem hledal kabely. Další krabice obsahovala spoustu starých kondenzátorů s logem TESLA Strašnice a. s. a spoustu obvodů NE555. Vedle dvou krabic plných směsí odporů, tranzistorů (samozřejmě taky TESLA) a 12V žárovíček stála další krabice se spoustou sklářského nářadí a brožurkou „Jak si opravit elektronku doma“ psanou na psacím stroji.

V poslední krabici jsem našel akorát ručně psaný sešit „Uzly v Systému“. Na každém řádku byl nějaký podivný název a latence.

Žádná paměť tu není. Co kdybych ale ukládal data do sítě tím, že je pošlu na nějaký vzdálený uzel? Než se mi vrátí, tak mohu volné místo využít na něco jiného. No a až se mi vrátí, tak je pošlu znovu. A takhle mohu posílat data na více uzlů zároveň. Jak to ale udělat co neefektivněji?

Změřil jsem si dobu odezvy jednotlivých uzlů a teď už mi jen stačí vybrat na které data posílat, aby můj uzel zvládl uložit co nejvíce dat.

Tvůj uzel komunikuje s uzly podle předem nastaveného rozvrhu. Rozvrh se sestává z k slotů, každý slot je jednu milisekundu dlouhý. V každém slotu může tvůj uzel komunikovat právě s jedním konkrétním uzlem a každému uzlu lze v rozvrhu přiřadit nejvýše jeden slot. Rozvrh se donekonečna beze změn opakuje.

Prvních k milisekund musí být plně zaplněno vysíláním dat ke k různým uzlům (různé uzly mohou mít stejnou frekvenci), tím se vyberou uzly a přidělí jejich slot. Žádné jiné uzly už se dále komunikace nesmí účastnit. V dalších opakováních rozvrhu je možné čekat na data, která nemusí přijít. Data se

ale nesmí vrátit mimo slot vyhrazený uzlu, na který byla vyslána, to by vedlo ke ztrátě dat.

Vstup

Vstup začíná řádkem s číslem $t \leq 20$, které udává počet zadání. Každé zadání začíná řádkem s číslem n ($1 \leq n \leq 10^4$), udávající počet uzlů, ke kterým můžeš poslat balík dat. Po něm následuje řádek s n kladnými celými čísly, představujícími latenci jednotlivých uzlů.

Po odeslání dat na uzel s latencí m ($1 \leq m \leq 10^7$), ti od uzlu po m milisekundách dorazí data zpět a ty je ve stejné milisekundě znovu odešleš na ten samý uzel. Pokud například odešleš data 5 milisekund od začátku vysílání na uzel s latencí 2, data se ti vrátí v 7. milisekundě, ihned se znovu odešlou, aby se zase vrátily v 9. milisekundě, odeslaly, vrátily, atd.

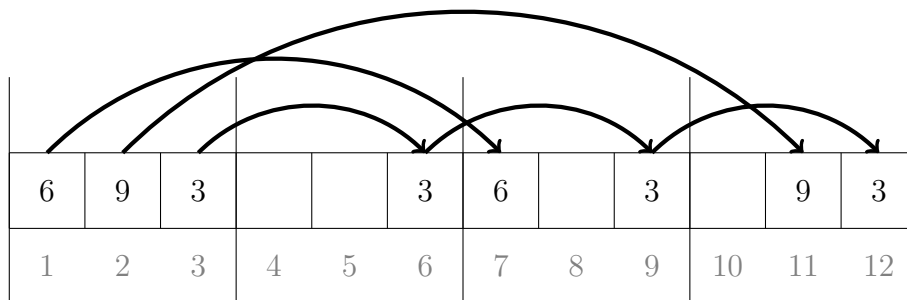
Výstup

Na výstup vypiš pro každé zadání jeden řádek s co největším možným číslem k reprezentující délku rozvrhu.

Příklady

Pro latence 6, 9, 3 je rozvrh pro $k = 3$ validní (Obrázek 1), protože

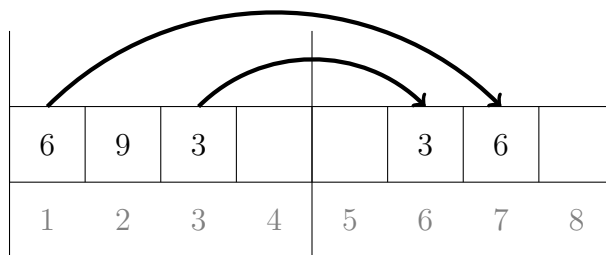
1. první tři sloty jsou plně obsazené,
2. každý uzel má přiřazený stejný slot v každé periodě rozvrhu (např. uzel s latencí 3 se v každé periodě vyskytuje výhradně na pozici 3, uzel s latencí 9 výhradně na pozici 2 a uzel s latencí 6 výhradně na pozici 1)



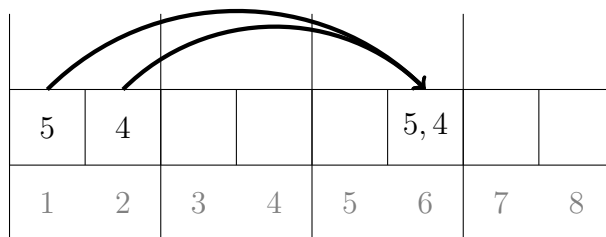
Obrázek 1: Validní rozvrh pro $k = 3$ a uzly s latencemi 6, 9, 3

Rozvrh na Obrázku 2 validní není, protože čtvrtý slot v prvních 4 slotech není obsazený. Navíc uzel s latencí 3 okupuje v první periodě pozici 3, ale ve druhé periodě pozici 2, tedy tohle taky porušuje požadavky.

Na obrázku 3 je rozvrh pro $k = 2$ pro uzly s latencemi 4, 5 nevalidní, protože se ve třetí periodě potkají na stejné pozici (nastává kolize).



Obrázek 2: Nevalidní rozvrh pro $k = 4$ a uzly s latencemi 6, 9, 3



Obrázek 3: Nevalidní rozvrh pro $k = 2$ a uzly s latencemi 4, 5

Ukázkové vstupy

Vstup

3
 8
 4 7 3 2 9 6 5 7
 10
 1 2 2 3 3 4 4 4 8 9
 10
 5 5 5 5 5 5 5 5 5 5

Výstup

3
 4
 5