

fiks!



ČESKÉ
VYSOKÉ
UČENÍ
TECHNICKÉ
V PRAZE

FIT

Fitácký Informatický Korespondenční Seminář

Ročník 2023/24, 1. kolo

Co je to FIKS?

FIKS je Fiťácký Informatický Korespondenční Seminář pro středoškolské studenty pořádaný Fakultou informačních technologií ČVUT v Praze. Byl založen na podzim roku 2013 a nyní tak probíhá desátý ročník (samozřejmě číslujeme od nuly). Nabízí možnost potrápít tvůj mozek řešením algoritmických úloh různé obtížnosti, od snadných po zapeklité, na nichž se můžeš leccos nového naučit a podstatně se zdokonalit.

Jak to probíhá?

Jeden ročník se skládá z několika kol a následného soustředění pro nejlepší řešitele. V těchto kolech máš možnost v teple domova řešit zadané úlohy, a své řešení nám potom odešleš. My ti toto řešení opravíme, obodujeme a pošleme zpět, aby ses mohl poučit ze svých chyb. Spolu s tím zveřejníme vzorové řešení, které můžeš prostudovat a třeba se něco přiučit. Získané body se sčítají do konečného žebříčku, ze kterého vybereme ty nejlepší a pozveme je na již zmíněné soustředění.

Proč řešit FIKS?

Řešením každého problému, se kterým se potýkáme, se zdokonalujeme. Zde ti nabízíme možnost pořádně se zamyslet nad zajímavými algoritmickými problémy, vyzkoušet své algoritmické myšlení a programátorské dovednosti a naučit se spoustu nových věcí.

Také je to možnost seznámení s novými lidmi, které baví informatika, programování, matematika a přemýšlení vůbec. Těm nejlepším jsme schopni garantovat přijetí na FIT ČVUT bez přijímacích zkoušek.

Jak se můžu zapojit?

Začni nejprve tím, že se zaregistruješ na našich webových stránkách na adrese <https://fiks.fit.cvut.cz>. Potom si stáhni zadání úloh (nebo využij tuto brožurku), vyřeš je a své řešení nám tamtéž odevzdej.

Typy úloh

Celkem se ve FIKSu můžeš setkat se třemi typy úloh. O který typ úlohy se jedná, je vždy uvedeno u konkrétního zadání úlohy.

Nejčastěji se u nás potkáš s úlohami typu *Rozmysli, popiš a naprogramuj*. U každé úlohy tohoto typu se odevzdává jak popis algoritmu (s odhadem asymptotické složitosti), tak i zdrojový kód řešení problému v tebou zvoleném jazyce (jakýkoliv vyšší programovací jazyk dle tvé volby, například C, Java, Pascal, apod.).

Dalším typem jsou úlohy *Zamysli se*. Tyto úlohy jsou obvykle více teoretické a vyžadují, aby ses nad nimi důkladně zamyslel. Oproti předchozímu typu úloh nemusíš nic programovat, odevzdává se pouze slovní popis řešení problému.

Pokud nemáš rád teoretické úlohy a raději by sis procvičil/a své programátorské umění, pak pro je pro tebe určena kategorie *Odpověz Sfinze*. V úlohách tohoto typu

po tobě nechceme popis algoritmu, je však potřeba vyřešit daný problém a toto řešení pak precizně naprogramovat. Oproti ostatním typům úloh se navíc okamžitě dozvíš, zda je tvé řešení správné, protože ho můžeš okamžitě odevzdat do našeho vyhodnocovacího systému.

Další a podrobnější informace nalezneš na našich webových stránkách.

Milý řešiteli FIKSu!

Další ročník FIKSu je tady! Jako vždy se můžeš těšit na spoustu úloh, které potrápí tvé mozkové závity a na kterých si procvičíš své algoritnické dovednosti.

V tomto PDF vidíš brožurku obsahující zadání 1. kola letošního ročníku FIKS. Pokud jsi nový řešitel, tak nezoufej, na předchozí stránce si můžeš přečíst základní informace, které k řešení úloh budeš potřebovat. V této brožurce naleznáš v kompaktní formě všechna zadání pro 1. kolo.

Tento ročník se skládá z pěti kol, jejichž přesné datum naleznáš na našem webu. V každém kole na tebe čekají tři úlohy, přičemž každá úloha je za 10 bodů (při odevzdání v termínu kola). Celkový počet získatelných bodů je tedy 150. Pro získání titulu 'úspěšný řešitel' je potřeba získat polovinu všech možných bodů (tj. 75 bodů) a zároveň získat alespoň 10 bodů z úloh typu 'Rozmysli, popiš a naprogramuj!' nebo 'Zamysli se!', kterých bude v součtu minimálně pět. Není však zaručeno, že v posledním kole bude nějaká z úloh těchto typů.

Nezbývá než ti popřát hodně štěstí při řešení úloh prvního kola. A nezapomeň – nejlepší řešitelé si vyslouží pozvánku na tradiční jarní soustředění! Tak hurá do toho!

Tvoji organizátoři

Fíťácký Informatický Korespondenční Seminář Ročník 2023/24, 1. kolo

Začátek kola:	23.9.2023 00:00
Termín odevzdání	27.10.2023 23:59
Zveřejnění výsledků:	27.11.2023 23:59
Odevzdávání:	Přes webové rozhraní na https://fiks.fit.cvut.cz
Další informace:	https://fiks.fit.cvut.cz kontakt@fiks.fit.cvut.cz
Discord:	https://discord.com/invite/Pc2w7hp

„Dobrý den, prosím vás, kde najdu DOD FITu?“ Vyhublý pán ve vybledlém tričku s logem Windows 95 se pomalu, jakoby unaveně, otočil. Jeho řídké vousy pokrývaly velkou část jeho staré vrásčité neudržované tváře. Jeho temné oči probodly mé „I use Arch BTW“ tričko, vzhledly do mé tváře a hned uhly pryč, aby nechaly mozkou prostor pro přemýšlení. Proboha, kam jsem se to zase dostal, pomyslel jsem si. Zíral do prázdna. Snad proto, že jsem ho vyrušil. Pak se zdánlivě ušklíbl a podíval se mi znovu do očí. „Pojďte, ukážu vám cestu,“ řekl a vyrazil. Asi mnou pohrdá. Bůhví kolik lidí sem už zabloudilo.

„Tady je respirium, studenti sem chodí odpočívát,“ řekl v otočení a vypadal jako by se snažil usmát. Jak zjistil, že to nejde, cukl tváří zpět a pokračoval v chůzi ignoruje moje zvukné, jakoby přátelské, přikývnutí.

Na konci chodby otevíraje dvoukřídlové dveře vedle, nyní již zeleně svítící, zaprášené čtečky karet řekl: „Vezmeme to zkratkou.“ Tentokrát se již úsměv předstírat ne-snažil. Před námi se otevřela dlouhá chodba, která narušovala mé eukleidovské chápání prostoru. Byli jsme přece několik pater nad zemí a takhle široká budova nevypadala.

„Pěkné to tady na FITu máte,“ řekl jsem snaže se zahájit konverzaci. „Tady jste stále ještě u architektů,“ odsekl. „Ale vy jste informatik, ne?“ „Ano... Ale na architektuře.“ „Správce sítě?“ „Něco takového.“ Začal jsem chápat jeho unavené oči a zašlou tvář. „Jak starou techniku tady máte?“ zeptal jsem se snaže se vyjádřit soustrast. „Klidně vám ji ukážu,“ řekl a na tváři se mu objevilo něco, co vzdáleně připomínalo úsměv.

Došli jsme na konec dlouhé chodby, kde byly jediné dveře, které otevřel. Pokynul mi dovnitř a já poslechl.

Úloha č. 1

Euklidovská kabeláž



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“.*

Potemnělá místnost měla velmi nízký strop, asi jen dva metry. Zato byla prostorná. Přes hromady krabic a kabelů jsem někde v dálce tušil zeď ztrácející se v oblacích prachu. Po obvodu místnosti byly racky zbastelené ze všech materiálů, které drží tvar, a blikající světélka různých zdrojů. Mezi LED diodami prokukovaly 12V žárovky a o kousek dál jakoby vycházela záře elektronek. Světlo indikačních světélek se mísilo se září sedmi-segmentových displejů, znakových LCDček a znakových výbojek. Uprostřed místnosti stál CRT monitor s blikajícím kurzorem terminálu, který osvětloval prach vířící se ve vzduchu.

Jak jsem tam v úžasu stál, skoro jsem nezaregistroval, že se za mnou dveře zavřely. Zůstal jsem sám. Tma a prach mě pohltily. Na terminálu se vypsalo „Neboj, Frodo. Správce není zlý. Jenom už si neví rady. Jeho síla slábne a už si se Systémem neví rady. Potřebuje někoho mladého a v tobě ucítil naději. V Systému se teď dějí strašné věci. Pomoz. [Y/n]“

Stiskl jsem klávesu Y.

„Nejdříve je potřeba udělat pořádek v kabeláži. Všechna zařízení v místnosti je potřeba propojit. Je jedno, jak to uděláš, ale každé zařízení musí být připojené k jednomu dalšímu. Aby byl ale v kabelech pořádek, můžeš je vést pouze po zemi a nesmí se křížit.“

Porozhlédl jsem se po místnosti. Všechna zařízení byla po jejím obvodu a každé zařízení mělo právě jednu zdířku, kterou ho šlo propojit s jiným.

Teď už jen vyřešit kabely. Po místnosti se jich našťestí válely plné krabice. Jenom jejich stav nebyl dvakrát zárný. Měly všechny možné koncovky a vypadaly, jako by zažily i zárné časy Fortranu. Jejich izolace byla popraskaná, v mnohých místech děravá a kov kabelů byl vidět skrz díry. Všechny kabely ale měly jednu vlastnost. Vždy měly obě koncovky stejné.

Vstup

Na prvním řádku se nachází přirozené číslo t ($1 \leq t \leq 1000$), které udává udává počet zadání, které následují.

Na prvním řádku každého zadání se nachází mezerou oddělená tři čísla: h, w, n ($1 \leq h \leq 10000$, $1 \leq w \leq 10000$, $1 \leq n \leq 40000$). Číslo h značí délku místnosti, w značí šířku místnosti a n značí počet zařízení. Následuje n řádků s informacemi o každém zařízení. Každý řádek obsahuje mezerou oddělené tři údaje y, x, c (číslo, číslo a řetězec písmen) ($0 \leq y \leq h, 0 \leq x \leq w, 1 \leq |c| \leq 100$). Čísla y a x společně značí lokaci zařízení v místnosti, kde y udává souřadnici vzhledem k délce a x vzhledem k šířce. Levý horní roh odpovídá souřadnici $(y, x) = (0, 0)$ a pravý dolní roh odpovídá souřadnici $(y, x) = (h, w)$. Za souřadnicemi se nachází řetězec znaků c o délce $|c| \leq 100$ udávající konektor zařízení (tedy všechna jména zařízení mají maximálně 100 znaků). Toto zařízení může být propojeno pouze s jedním zařízením stejného konektoru. Můžete se spolehnout, že všechna zařízení se nachází jen po obvodu místnosti. Tedy formálně pro každé zařízení na pozici (y, x) platí alespoň

jedna z následujících čtyř podmínek: $y = 0, x = 0, y = h, x = w$. Dále se můžete spolehnout na to, že žádné dvě zařízení se nenachází na shodných souřadnicích.

Výstup

Vášim úkolem je pro každé z t zadání zahlásit na výstup `pujde to` pokud lze v místnosti propojit kabely tak, že se žádné dva kabely nekříží a všechny zařízení jsou propojeny s některým ze zařízení stejného konektoru. Pokud toto nelze, zahlaste na výstup `ajajaj`.

Vstup

```
2
2 3 4
0 1 DVI
0 2 JACK3MM
2 1 DISPLAYPORT
2 0 SATA
2 3 2
1 0 HDMI
2 1 HDMI
```

Výstup

```
ajajaj
pujde to
```

Vstup

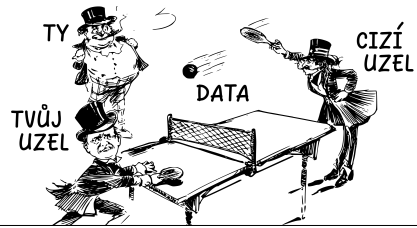
```
2
2 3 4
1 3 VGA
0 0 USB-A
0 1 VGA
2 0 USB-A
2 3 4
2 3 ETHERNET
1 0 ETHERNET
0 1 USB-C
2 2 USB-C
```

Výstup

```
pujde to
ajajaj
```

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 zavadila 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 přehrátké komponenty, na kterých se se syčením měnila v páru. Asi vodní chlazení.

Pára místic 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 zároviček 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 uzlu? 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 nejefektivněji?

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

Tvůj uzlu komunikuje s uzly podle předem nastaveného rozvrhu. Rozvrh se sestává z k slotů, každý slot je jednu milisekundu dlouhý. V každém slotu může tvůj uzlu 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 můžou mít stejnou frekvenci), tím se vyberou uzly a přidělí jejich slot. Žádné jiné uzly už se dále komunikace nesmí účastnit. V dalších opakovaných 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.

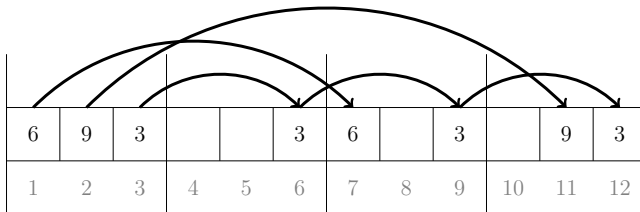
Výstup

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

Příklady

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

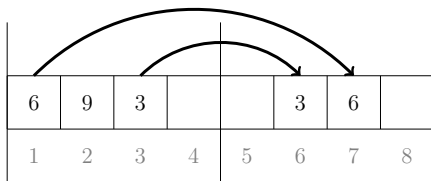
1. první tři sloty jsou plně obsazené,
2. každý uzel má přiřazený stejný slot v každé periodě rozvrhu (např. uzel s latencí 3 se v každé periodě vyskytuje výhradně na pozici 3, uzel s latencí 9 výhradně na pozici 2 a uzel s latencí 6 výhradně na pozici 1)



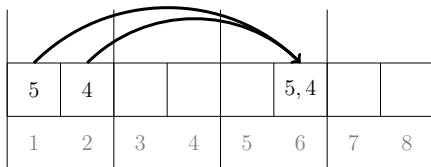
Obrázek 1: Validní rozvrh pro $k = 3$ a uzly s latencemi 6, 9, 3

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

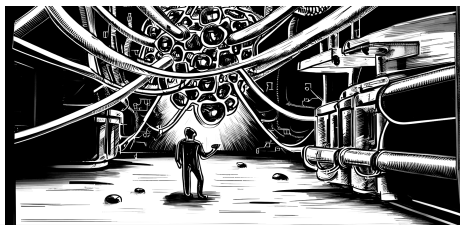
3
 8
 4 7 3 2 9 6 5 7
 10
 1 2 2 3 3 4 4 4 8 9
 10
 5 5 5 5 5 5 5 5 5

Výstup

3
 4
 5

Úloha č. 3

Meziuzelná autorizace



Zamysli se!

10 b

Tato úloha je čistě teoretická, tvým úkolem zde není napsat program. Namísto toho si dej záležet na kvalitním slovním popisu, kde mimo jiné jasně zdůvodníš, proč tvůj postup skutečně bude fungovat.

S dostatkem paměti proces pokračoval dál. Místnost se plnila prachem a mlhou, až v ní nebylo vidět na krok. Hukot větráků sílil, relátka cvakala jako o život a voda syčela, jak se komponenty přehřívaly. Zář po obvodu místnosti mě oslňovala. Různé zvuky se už úplně slily v jeden souvislý a hlasitý hukot. Světlo a hukot se už nedaly vydržet. Zavřel jsem oči. Zakryl si uši. „Aaaaaa“, zakřičel jsem. Všechno ztichlo.

Otevřel jsem oči. Mlha byla pryč. Prach zmizel. Všechny kabely zůstaly na svém místě, ale nestál jsem přímo na nich. Dotkl jsem se země. Jako by byly zalité v pryskyřici a já chodil nad nimi. Stejně tak zařízení po obvodu místnosti. Přesto fungovala dál. Všechny displeje a kontrolní LED diody normálně fungovaly, jenom hukot zmizel. Vše se zdálo být v kruhovém prostoru, jak jsem vytušil, když jsem nemohl najít rohy místnosti. Nebo vlastně uzlu? Nevím. V místě CRT monitoru se vznášel obraz ve vzduchu.

V rohu obrazu bylo napsáno „Injection authorization required“. Otevřel jsem manuálovou stránku „injection“.

injection - proces přenosu živých bytostí do Systému, opak surjekce

Varování: Na proces injekce doporučujeme ochranu očí a uší.

Po úspěšné injekci je potřeba dostat autorizaci od ostatních uzlů, aby mohla probíhat meziuzelná komunikace.

Autorizaci provedete dotazáním uzlů ve vašem okolí. Aktuálně jsou používané dva protokoly - Standard Authorization Protocol a Microsoft Authorization Protocol. Oba protokoly dělají prakticky to stejné, jenom MAP obsahuje pár odlišností, aby byl nekompatibilní se SAP.

Pro získání autorizace je potřeba poslat uzlu žádost, která musí obsahovat v hlavičce identifikátor protokolu. Pokud pošlete žádost se špatným identifikátorem, uzel si vás zapamatuje a žádosti o autorizaci nikdy nevyhoví.

Ke zjištění, o který protokol jde, lze použít trik. Oba protokoly obsahují experimentální modul na obecné dotazování. To vám umožňuje se ptát uzlů na libovolné otázky, na které lze odpovědět jedním bitem (tedy ANO/NE). Navíc se lze ptát více uzlů a odpověď se zagreguje. Tento modul funguje v obou protokolech stejně, liší se pouze jedním implementačním detailem. MAP má výslednou hodnotu bitu invertovanou. Odpovídá tedy obráceně než by člověk očekával u SAPu.

...

Prošel jsem i zbytek manuálu. V ničem jiném se tyto dva protokoly neliší. Zadal jsem tedy dotaz na nějaký uzel.

You asked 1 node. This number of nodes is currently not supported. Supported number of nodes is: 3

Zadal jsem tedy stejný dotaz pro BÚNO první tři uzly.

Majority of nodes in this group answered: YES

Jak na co nejmenší počet pokusů zjistím správný protokol pro každý uzel?

Rozšíření

Některé nody se za svůj protokol stydí, takže pokud s nimi není ve skupině někdo, kdo používá stejný protokol a nestydí se za něj, tak odpovídají opačně než jak by odpověděli, kdyby byli sami. Pokud však vybereš skupinku, kde budou všechny nody používat stejný protokol, za který se stydí, tak se nemají před kým stydět a odpoví jak se na jejich protokol sluší.

Shrnutí autorizačního procesu

- Každý protokol je používán alespon třemi uzly.
- Můžeš se ptát na jakoukoli otázku, na kterou je odpovědí ANO nebo NE.
- Ptáš se vždy trojice uzlů, odpověď pro tebe bude taková, jaká převažovala mezi uzly. Ty se nedozvíš, co odpověděly jednotlivé uzly.
- Uzly od sebe zvládneš rozeznat a trojici, které se zeptáš, si můžeš vybrat.
- Uzly pomocí svého interního API ví, jaký protokol používají ostatní uzly.
- Tvým úkolem je použít co nejméně dotazů, abys každému uzlu mohl poslat žádost o autorizaci (musíš vědět správný protokol).

Ukázkové dotazy

Dotaz	1 + 1 = 2	1 + 1 = 3
SAP, SAP, MAP	ANO	NE
MAP, MAP, MAP	NE	ANO

Ukázky s rozšířením

Dotaz	1 + 1 = 2	1 + 1 = 3
shy SAP, shy SAP, MAP	NE	ANO
shy SAP, SAP, MAP	ANO	NE
shy SAP, shy SAP, shy MAP	ANO	NE
shy SAP, shy SAP, shy SAP	ANO	NE