

Milý řešiteli,

vítáme Tě u 2. série úloh 6. ročníku korespondenčního semináře MoRoUS. Jako v minulých letech i nyní Ti přinášíme úlohy ze světa profesora Morouse a jeho robotických společníků.

Řešení úloh Ti umožní nejen nahlédnout do zajímavých zákoutí umělé inteligence a robotiky, ale také zúčastnit se jarního soustředění s inspirativními lidmi, obzor rozšiřujícími přednáškami a nezapomenutelnými zážitky ;) A na nejlepší řešitele čekají na konci soutěže tematické ceny!

2. série 2019/2020

Termín odeslání 2. série: **26. 1. 2020**

Kam posílat řešení?

Až budeš mít řešení hotové, pošli nám je včetně všech nákresů, programků a prostě všeho co by nám usnadnilo opravování Tvé úlohy. Stačí, když pošleš řešení jen některých úloh nebo jejich částí, i k těm Ti pošleme komentáře a přičteme za ně body ;)

Řešení posílej e-mailem na adresu seminar@morous.fel.cvut.cz

Úloha č. 1: (20 bodů)

Profesor Morous se díval na přistávací dráhu, na kterou pomalu dopadal první sníh... A také poslední dopravní letoun tohoto roku. Výborně, nová zásilka thronia je tady, pomyslel si, když viděl oranžové bedny, které jeho roboti vynášeli z nákladního prostoru. To je dobře, že stihla dorazit ještě před svátky, náhradní materiál už jsme opravdu potřebovali!

Dole na rampě ale taková idyla nevládla. Kam s tím? Všechny sklady už jsou plné. „To bude tím, že tam všechno házete jen tak hala bala bez ladu a skladu,“ prohlásil robot velící směně. „To se pak jeden nemůže divit, že se tam nevejde pár koulí navíc... Pusíte se do úklidu a alespoň ten nový materiál uložte co nejúspěšněji!“

Leckoho by taková úloha znechutila, ale protože byli roboti navrženi tak, aby je optimalizační úlohy opravdu bavily, rozběhly se teď jejich obvody naplno.

„Jak asi nejlépe tam ty kulové balíky naskládáme?“ uvažoval nahlas Izák (zvukový výstup byl defaultně zapnutý, aby se dalo kontrolovat, co robot zrovna provádí).

„To je taky nápad, balit materiál do tvaru, který nejméně efektivně využívá prostor...“ rozčiloval se Euklid. „Kdybychom alespoň pracovali v jiné dimenzi!“

„Jo, to by se ti líbilo,“ uchechtl se René. „Třeba jen jedna, to by ti stačilo? Takhle se vyhýbat problémům...“

Úkol 1.1

Začneme zlehka a budeme uvažovat jen jednodimenzionální prostor (přímku). Jak na ni co nejúspěšněji naskládat jednodimenzionální „koule“? Zvaž nejprve, jak taková koule bude vypadat. Jaké bude v tomto případě procento zaplnění prostoru?

Úkol 1.2

Trošku přitvrdíme – tady už možná budeš potřebovat tužku a papír – dostáváme se do dvoudimenzionálního prostoru. Představ si pokrytí tohoto prostoru koulemi, jejichž středy jsou uspořádány pravidelně ve čtvercové mřížce. Jaké bude procento zaplnění prostoru nyní? (vypočítáš jej jako podíl objemu zaplněné části a objemu celého prostoru, nejlépe na nějaké pravidelně se opakující části prostoru). Existují i efektivnější uspořádání koulí ve 2D. Které myslíš, že je nejlepší a jaký má koeficient zaplnění?

Úkol 1.3

A konečně naše realita. Jak efektivní bude uspořádání koulí se středy ve čtvercové mřížce ve 3D? Jaká vymyslíš další (efektivnější) uspořádání a jaká mají koeficient zaplnění?

Úkol 1.4

Přece bychom se nespokojili jen se třemi dimenzemi. Jak se bude koeficient zaplnění měnit s přibývajícimi dimenzemi, použijeme-li pro výpočet obsahu koule o poloměru R následující vzorce?

$$V_{2k}(R) = \frac{\pi^k}{k!} R^{2k}$$

pro sudé dimenze ($D = 2k$)

$$V_{2k+1}(R) = \frac{2(k!)(4\pi)^k}{(2k+1)!} R^{2k+1}$$

pro liché dimenze ($D = 2k + 1$)

(Více na https://en.wikipedia.org/wiki/Volume_of_an_n-ball)

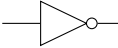

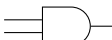



Uvažuj opět uspořádání středů do čtvercové mřížky a uveď řešení alespoň pro prvních pět dimenzí.

Symbol vykřičníku je znakem pro *faktoriál*, který je definován jako $n! = 1 \cdot 2 \cdot \dots \cdot n$, tedy například $4! = 1 \cdot 2 \cdot 3 \cdot 4$. Faktoriál nuly a jedné má přiřazenu hodnotu jedna. (Více na <https://cs.wikipedia.org/wiki/Faktoriál>)

Úloha č. 2: Z kroniky doktora Daystrůma (20 bodů)

V minulém díle

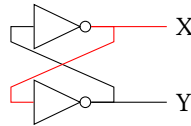
Doktor Daystrům již zná následující jednoduché součástky. Od teď jim můžeme říkat také **logická hradla**.

<table border="1" style="display: inline-table; vertical-align: middle;"> <thead> <tr><th>A</th><th>X</th></tr> </thead> <tbody> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </tbody> </table>	A	X	0	1	1	0	<p>ICKE</p> 	<table border="1" style="display: inline-table; vertical-align: middle;"> <thead> <tr><th>A</th><th>B</th><th>X</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	X	0	0	0	0	1	1	1	0	1	1	1	1	<p>ELLER</p> 									
A	X																																
0	1																																
1	0																																
A	B	X																															
0	0	0																															
0	1	1																															
1	0	1																															
1	1	1																															
<table border="1" style="display: inline-table; vertical-align: middle;"> <thead> <tr><th>A</th><th>B</th><th>X</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	X	0	0	0	0	1	0	1	0	0	1	1	1	<p>OCH</p> 	<table border="1" style="display: inline-table; vertical-align: middle;"> <thead> <tr><th>A</th><th>B</th><th>X</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	X	0	0	1	0	1	0	1	0	0	1	1	0	<p>NELLER</p> 
A	B	X																															
0	0	0																															
0	1	0																															
1	0	0																															
1	1	1																															
A	B	X																															
0	0	1																															
0	1	0																															
1	0	0																															
1	1	0																															
<table border="1" style="display: inline-table; vertical-align: middle;"> <thead> <tr><th>A</th><th>B</th><th>X</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	X	0	0	1	0	1	1	1	0	1	1	1	0	<p>NOCH</p> 	<table border="1" style="display: inline-table; vertical-align: middle;"> <thead> <tr><th>A</th><th>B</th><th>X</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	X	0	0	0	0	1	1	1	0	1	1	1	0	<p>AELLER</p> 
A	B	X																															
0	0	1																															
0	1	1																															
1	0	1																															
1	1	0																															
A	B	X																															
0	0	0																															
0	1	1																															
1	0	1																															
1	1	0																															

Jejich kombinací zvládne vyrobit všelijaké chytré obvody, dále též **kombinační obvody**. Jedním z takových kombinačních obvodů je například půl-sčítačka nebo sčítačka. Půl sčítačka je jednoduchý obvod, který umí sečíst dva bity (dvě jednobitová čísla). Výsledkem takového sčítání je pak dvoubitové číslo (nebo jeden bit výsledku a jeden bit přenosu). Sčítačka je pak vylepšení půl-sčítačky, která umí při sčítání dvou čísel zohlednit i přenos z nižšího řádu (tj. sečte tři bity, výsledek je opět dvoubitový).

Pokračování příběhu

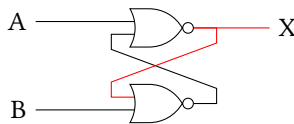
Jak si doktor Daystrům postupně pohrával s jednotlivými prvky, omylem se mu podařilo zapojit dvě součástky do kruhu. Obvod lehce zablikal a po chvíli se na výstupu X objevilo světlo a výstup Y byl zhasnutý. Daystrům obvod několikrát zcela rozebral a zase složil. Pokaždé obvodu chvíli trvalo, než se stav ustálil a pak se právě jeden z vývodů rozzářil. To, jestli svítil vývod X nebo Y záleželo na tom, která součástka zrovna byla rychlejší, jakým způsobem provedl zapojení a zda v propojkách bylo ještě nějaké zbytkové světlo.



To, že se obvod uměl ustálit a „pamatoval“ si svůj stav se Daystrómovi líbilo, jen nebyl nadšený z toho, že se obvod po stabilizaci nedal nijak ovlivnit. V tom ho ale napadlo, že by mohl vyzkoušet vyměnit hradla. Postavil podobný obvod, tentokrát ale hradla IČKE vyměnil za hradla NELLER.

Úkol 2.1

Popiš, jak se takový obvod bude chovat. Napiš pro tento obvod pravdivostní tabulku.



Pozn.: Výstup X je zároveň i vstupem. Tabulka tedy má záhlaví:

A	B	X	X
-----	-----	-----	-----

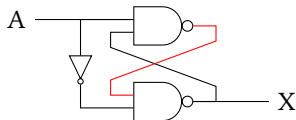
Úkol 2.2

Sestav obvod, který bude splňovat následující pravdivostní tabulku.

A	B	X	X
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Obvod se tedy chová tak, že pro zhasnuté A a B si „pamatuje“ svůj stav. Když jsou oba vstupy rozsvícené, obvod se přepíná. Samostatně vstup A funguje jako nastavovač hodnoty (svítí-li, bude svítit i výstup), vstup B pro změnu resetuje výstup do nerozsvíceného stavu.

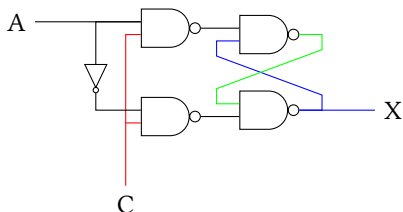
Čím více si Daystrům s obvodem hrál, tím více mu přišlo nepraktické používat dva různé vstupy. Provedl tedy malou změnu v jednom ze svých obvodů a oba vstupy spojil skrze hradlo ICKE.



Úkol 2.3

Zapsat pravdivostní tabulku obvodu a popsat chování pro Tebe jistě bude hračka.

Doktor Daystrům si najednou uvědomil, že tyto typy obvodů jsou po celou dobu velmi citlivé na vstupní hodnoty. Jeho poslední obvod je velmi šikovný, ale občas by se hodilo mu vstupy na chvíli „vypnout“, aby světlo na výstupu obvodu zběsile nelítalo, když se na vstupu něco neplánovaně změní. Přidáním nového vstupu C a hradla ke každému z původních vstupů získal mnohem chytrější obvod.



Pokud Daystrům do vstupu C posvítí, obvod se chová identicky, jako v předchozím případě. Je-li C zhasnuté, obvod zcela ignoruje změny na vstupu A. Doktor se zaradoval, pokud tuto nastavbu přidá do všech obvodů, může jejich funkci signálem C synchronizovat.

Úkol 2.4

Občas se ale může naopak hodit možnost nastavit vnitřní hodnotu obvodu nezávisle na synchronizačním vstupu. Vymysli, jak obohatit obvod, aby se dala nastavit vnitřní hodnota nezávisle na vstupu C (jeho současná funkce ale musí zůstat zachována).

To bude pěkný základ pro můj nový paměťový obvod, pomyslel si Daystrům a ihned ho napadla aplikace.

Úkol 2.5

Navrhni obvod, který si bude umět „zapamatovat“ dvoubitové číslo (reprezentované pomocí dvou jednobitových výstupů X a Y) a pomocí dvou řídicích vstupů A a B bude umět po jedničce přičítat, resp. odčítat.

Pozn.: Pro jednoduchost můžeme předpokládat, že celý obvod bude řízený hodinovým vstupem C , do kterého se například každou vteřinu na chvilíčku zasvítí. Bude-li v ten okamžik vstup A i B zhasnutý, hodnota X a Y se nezmění. Pokud se do A v daném okamžiku svítí, hodnota čísla XY se zvětší o jedna (00, 01, 10, 11, 00, ...). Pokud se svítí do B , hodnota se zmenší o jedna (11, 10, 01, 00, 11, ...). Situaci, kdy se svítí do vstupu A i B by neměla nastat a není tak nutné ji uvažovat, obvod se může chovat, jak se mu zlíbí.