

2021 Aplikačné program... / Cvičenia



1 Sudoku



Vytvoril/a Peter Koscelanský [Administrator]

Posledná aktualizácia: Oct 05, 2021

Vytvorte konzolovú aplikáciu, ktorá bude mať dva voliteľné parametre

- `-i` vstupný súbor
- `-o` výstupný súbor

Ak vstupný, alebo výstupný súbor nebude uvedený použije sa štandardný vstup, respektíve výstup. Napríklad

	Príkazový riadok	Vstup	Výstup	Pozn
1	<code>-i input.txt -o output.txt</code>	Súbor input.txt	Súbor output.txt	
2	<code>-o output.txt -i input.txt</code>	Súbor input.txt	Súbor output.txt	Nezál paran
3	<code>-i input.txt</code>	Súbor input.txt	Štandardný výstup	
4		Štandardný vstup	Štandardný výstup	Bez k
5	<code>-i -o output.txt</code>	N/A	N/A	Chyba nepoc
6	<code>-i -o</code>	Súbor -o	Štandardný výstup	Param začiat meno sprsuj

Aplikácia zo vstupu načíta po riadkoch Sudoku (formát .sdm). V riadku je práve

81 znakov, čo zodpovedá jednému 9x9 Sudoku zapísanému po riadkoch. Čísla 1-

9 zodpovedajú samotným vyplneným číslam, prázdne miesta sú buď ., alebo 0, ostatné znaky sú zakázané. Rovnako sa netrápte s koncami riadkov, spoľahnite sa, že to za vás vyrieši C++. Na výstup dá aplikácia jednotlivé vyriešené Sudoku v rovnakom poradí ako boli na vstupe (čiže 81 znakov 1-9, nebudú tam teda nevyplnené miesta). Ak náhodou zadanie Sudoku nemá riešenie, nechajte namiesto neho prázdny riadok. Pokiaľ nechcete bonusový bod, tak rýchlosť nie je potrebné riešiť. Stačí implementovať prehľadávanie všetkých možností s tým, že ich budete rezat' ak nie je niečo splnené podľa pravidiel Sudoku. Štandardne sudoku nie je platné, ak má viac ako jedno riešenie, toto nás netrápe, nato máme bonusový bod :).

Napríklad ak spustíme `sudoku.exe -i input.sdm -o output.txt` a vstupný súbor bude vyzerat'

```
1 0164000002000090004000000620702301001000000030030870409600000050008000
2 049008605003007000000000030000400800060815020001009000010000000006004
3 760500000000060008000000403200400800080000030005001007809000006000100
4 3164529782856793144973185628792341561429657836531872499687214955218436
5 000605000003020800045090270500000001062000540400000070980604500060407
6 40900070500001000000620780020000000900370420080000004002801500000600
7 0000100300400705010020080066800000030003020003000000452005008008010400
8 3164529782856793144973185628792341561429657836531872499687214355218436
9 0800700302600500180000004000006020003900100860007090000040008008100400
10 000093006008009000200061000000800530060002003700500000025000400010090
```

Potom výstupný súbor `output.txt` by mal byť

```
1 3164529782856793144973185628792341561429657836531872499687214355218436
2 1492386756239571487581462399354728614678159232813697543167945825926834
3 7635481294213697589581724632974368151867952343458216978192543766349175
4
5 8296753146731248951453982765874369219628175434319526873987614522165497
6 4196387257285196435362478912541863791937542688679231546428915373714659
7 7689154329432765815124387966851942731743529683296871452375698148517436
8 3164529782856793144973185628792341561429657836531872499687214355218436
9 4819762352674539189358214671786325493925147865467893217241658938193476
10 1852934766478159329237461852196847534563712983789526148925673415314298
```

ako vidieť na štvrtom riadku je sudoku, ktoré sa nedá vyriešiť (je už zle zadané),

preto ostal riadok prázdny.

Ak nastane akákoľvek chyba, teda napríklad zlý počet parametrov, neexistujúce súbory, alebo zlý formát súboru, skončíte s error kódom rôznym od `0`.

Bonusový bod sa dá získať za rýchlosť. Aplikáciám sa ako vstup dá súbor so stovkami/tisícami Sukodu a päť riešení s najrýchlejším (ale korektným 😊) spracovaním získajú bod.

Ešte jeden bonusový bod za podporu validácie sudoku. Ak dostanete parameter `-check`

`080070030260050018000000400000602000390010086000709000004000800810040052050090070`, tak na štandardný výstup vypíšete `0` ak sudoku nie je platné (teda buď nemá riešenie, alebo má viac ako jedno), `1` ak je Sudoku v poriadku. Tu pozor, lebo ak budete len tak skúšať všetky možnosti, tak vám to môže trvať veľmi dlho a my vždy máme časový limit.

👍 Páči sa mi to Bud'te prvý(-á), komu sa to páči

Žiadne označenia

💬 40 page comments



Anonymný Oct 10, 2021

Dobrý deň. Ak nie sú zadane žiadne parametre a teda pracujem so štandardným vstupom a výstupom, bude zadane v jednom spustení programu iba jedno sudoku? Alebo ich môže používateľ zadať na štandardný vstup viac, každé na nový riadok?

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 11, 2021

Viac, kazde na nový riadok.

Odpovedať • Páči sa mi to



Anonymný Oct 12, 2021

Zdravím,

čo má program spraviť, ak je vstup neplatný, teda v riadku nebude 81 znakov `0-9` a `.`, ale tam bude napríklad iba text `neplatny vstup` a koniec riadka?

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 12, 2021

Skončiť s nenulovým error kodom.

Odpovedať • Páči sa mi to



Anonymný Oct 13, 2021

Takže ak je na vstupe súbor, ktorý má prvý riadok platný, druhý neplatný a ďalších 5 riadkov platných, tak má vypísať riešenie na prvý riadok, a skončiť s nenulovým error kódom, alebo nevypísať nič a hneď skončiť s nenulovým error kódom?

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 13, 2021

Obe su dobre, hlavne o ten nenulovy error kod ide.

Odpovedať • Páči sa mi to



Anonymný Oct 13, 2021

Aj keď budú zvyšné vstupy platné? Napríklad by na 3. riadku ukážkovo vstupu bolo "neplatný vstup" ale ostatné sudoku by boli riešiteľné.

Odpovedať • Páči sa mi to



Anonymný Oct 12, 2021

Dobrý deň,

ak má sudoku dve a viac riešení, je jedno, ktoré vypíšeme?

Štandardne sudoku nie je platné, ak má viac ako jedno riešenie, toto nás netrápi, nato máme bonusový bod :).

Ďakujem.

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 12, 2021

Ano, je to jedno

Odpovedať • Páči sa mi to

**Anonymný** Oct 12, 2021

Dobrý deň,

pri načítaní zo štandardného vstupu, ako bude tento vstup ukončený? Resp. kedy máme ukončiť načítavanie nových sudoku.

Ďakujem.

Odpovedať • Páči sa mi to

**Peter Koscelanský [Administrator]** Oct 12, 2021

Dozvieš sa, ciatanie zo vstupu ti zlyha.

Odpovedať • Páči sa mi to

**Anonymný** Oct 12, 2021

Takze, ak spravne rozumiem, tak sa bavime o failbit-e?

Odpovedať • Páči sa mi to

**Peter Koscelanský [Administrator]** Oct 13, 2021

Pravdupovediac to neviem, radšej to kontrolujte iba ako

```
if (stream) { /*  
stream is OK */}
```

Odpovedať • Páči sa mi to

**Anonymný** Oct 12, 2021

Pri bonuse moze byt rovnako zadanych zo stdin viacero sudoku po sebe?

Napriklad pre platne, neplatne, platne, platne, neplatne bude vystup na stdout:

```
1 1  
2 0  
3 1  
4 1  
5 0
```

Dakujem.

Odpovedať • Páči sa mi to

**Peter Koscelanský [Administrator]** Oct 12, 2021

Nie, iba jeden.

Odpovedať • Páči sa mi to

**Anonymný** Oct 12, 2021

Dobrý deň,

Mal by som viacero otázok.

1. Aké znaky sú povolené pre meno vstupného/výstupného súboru a aké dlhé môže byť meno týchto súborov?
2. Čo ak je vstupný aj výstupný súbor rovnaký?
3. Môže byť viacero vstupných alebo výstupných súborov alebo stále len jeden vstupný, jeden výstupný?
4. Môžu byť v jednom zadanom riadku sudoku aj `.` aj `0` naraz alebo len jedno z toho?
5. Čo ak dostaneme vyriešené sudoku?

Za odpovede vopred ďakujem.

Odpovedať • Páči sa mi to

**Peter Koscelanský [Administrator]** Oct 12, 2021

1. Preto to riesis, zober to co je na vstupe a daj to do konstruktora i/ofstreamu.
2. TÁto je zaujimava, povedzme, ze je to nespecifikovane, teda, ze to moze urobit nieco definovane, ale inak hocico, mozes to detekovat, alebo proste otvorit jeden aj druhy subor a duham v najlepsie.
3. vzdy len jeden.
4. Vzdy len jeden.
5. TÁk to je super, to je sudoku, ktore sa da riesit a rovno mas aj riesenie. :)

Odpovedať • Páči sa mi to

**Anonymný** Oct 12, 2021

Cize toto je neplatny vstup a treba skoncit s nenulovym error kodom?

**306.529782856793144973185628792341561429657836531872499687214355
21843697734596821**

Dakujem

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 13, 2021

Ano je to neplatny vstup.

Odpovedať • Páči sa mi to



Lukáš Manduch Oct 13, 2021

Este ku tomuto príkladu, ak už niekto implementoval to, že bodka a nula môžu byť pomiesane tak budeme akceptovať aj to, netreba to prerábať.

Toto nebolo úplne v zadani špecifikované a nejdeme vám teraz meniť podmienky v polke týždňa.

Odpovedať • Páči sa mi to



Anonymný Oct 13, 2021

možeme používať `std::istream` namiesto `std::ifstream`?

Odpovedať • Páči sa mi to



Lukáš Manduch Oct 13, 2021

Môžete používať aj `istream`

Odpovedať • Páči sa mi to



Anonymný Oct 13, 2021

môžem používať `std::stack` ?

Odpovedať • Páči sa mi to



Lukáš Manduch Oct 13, 2021

Ano

Odpovedať • Páči sa mi to



Anonymný Oct 14, 2021

Ranko, chcela by som sa pýtať, že či je rozdiel keď v maine je `return 1` alebo v iných funkciách je `exit(1)`? dakujem

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 14, 2021

Tak rozdiel to samozrejme je, exit nevola destruktory lokálnych objektov. Ale z hľadiska výstupu programu je to jedno.

Odpovedať • Páči sa mi to



Anonymný Oct 14, 2021

Dobry den, chcel by som sa spytať ako má vyzerat výstup pri štandardnom vstupe, nakoľko tieto vstupy sú jedine čo mi nepresli a predpokladám že to bude kvôli formátovaniu.

Má to vyzerat takto, alebo majú byť prázdne riadky medzi vstupom/výstupom, prázdny riadok na konci, alebo ako?

```
016400000200009000400000062070230100100000003003087040960000005000800007000006
820
0490086050030070000000000030000400800060815020001009000010000000000600400804500
390
760500000000060008000000403200400800080000030005001007809000000600010000000003
041
316452978285679314497318562879234156142965783653187249968721495521843697734596
821
000605000003020800045090270500000001062000540400000007098060450006040700000203
000
409000705000010000006207800200000009003704200800000004002801500000060000905000
406
000010030040070501002008006680000003000302000300000045200500800801040020090020
000
316452978285679314497318562879234156142965783653187249968721435521843697734596
821
080070030260050018000000400000602000390010086000709000004000800810040052050090
070
000093006000800900020006100000080053006000200370050000002500040001009000700130
000
^Z
316452978285679314497318562879234156142965783653187249968721435521843697734596
821
149238675623957148758146239935472861467815923281369754316794582592683417874521
396
763548129421369758958172463297436815186795234345821697819254376634917582572683
941
829675314673124895145398276587436921962817543431952687398761452216549738754283
```


169

419638725728519643536247891254186379193754268867923154642891537371465982985372

416

768915432943276581512438796685194273174352968329687145237569814851743629496821

357

316452978285679314497318562879234156142965783653187249968721435521843697734596

821

481976235267453918935821467178632549392514786546789321724165893819347652653298

174

185293476647815932923746185219684753456371298378952614892567341531429867764138

529

Odpovedať • Páči sa mi to



Anonymný Oct 14, 2021

Pri standardnom vstupe aj výstupe* cize std::cin a std::cout

Odpovedať • Páči sa mi to



Anonymný Oct 14, 2021

Dobrý deň,

prechádza mi pri testovaní **Multiple cin cout**, ale pri **Simple cin cout** mi vyhadzuje chybu.

Ostatné testy sú všetky v pohode. Dost' dlho som sa s tým teraz trápil, ale nemôžem prísť na to, kde je chyba.

Veľmi pekne vopred ďakujem za odpoveď.

Odpovedať • Páči sa mi to



Anonymný Oct 14, 2021

Dobry vecer, ja sa pripajam s rovnakym problemom. 😞

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 15, 2021

Idem to pozrieť kde tam máte chybu, alebo možno máme chybu aj my. dnes poobede to bude vyriesene.

Odpovedať • Páči sa mi to



Anonymný Oct 15, 2021

Dakujem(e)

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 15, 2021

Je to fixnute, problem bol v tom, ze test pri konzole pocital s tym, ze po spracovaní jedného riadku vypisete rovno výsledok. (co bolo podľa mňa logickejšie ako to celé načítať do pamäte). Teraz to bude fungovať aj tak aj tak. Sorry za problémy. Ak vám teda nesli iba tieto dva testy pravdepodobne to máte dobre.

Odpovedať • Páči sa mi to



Anonymný Oct 15, 2021

V pohode, myslel som si že to bude tým, ale keď bolo povedané, že tester má oddelený vstup a výstup tak mi to nedávalo zmysel. Každopádne dakujem a skúsím otestovať znovu.

Odpovedať • Páči sa mi to



Anonymný Oct 14, 2021

Dobry vecer,

mame na vstupe ignorovať prázdne riadky? Teda myslím či toto je správny vstup:

```
1 0164000002000090004000000620702301001000000030030870409600000050008000070
2
3
4 0800700302600500180000004000006020003900100860007090000040008008100400520
```

a výstup má byť:

```
1 316452978285679314497318562879234156142965783653187249968721435521843697
2 4819762352674539189358214671786325493925147865467893217241658938193476520
```

alebo chyba a nenulový error kód?

Dakujem.

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 15, 2021

chyba 😊

Odpovedať • Páči sa mi to



Anonymný Oct 15, 2021

Dobry deň, chcela by som sa spýtať, či pri načítavaní zo štandardneho vstupu, teda std::cin, ho máme ukončiť pomocou ctrl+z alebo enterom navyše alebo nejako inak? Pýtam sa hlavne kvôli tým automatizovaným testom. Ďakujem.

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Oct 15, 2021

Myslím, že ctrl+z je standardný spôsob, ale tým sa az tak nemusíte zatazovať, vám proste raz bude std::cin hlasit false ak ho date do podmienky.

Odpovedať • Páči sa mi to



Anonymný Oct 15, 2021 ↗

Dobrý deň,

chápem správne, že v tomto zadání, ale aj pri budúcich zadaniach, môžeme používať celú škálu funkcií, ktoré nájdeme v std? Napríklad aj funkcie z <algorithm>? Alebo je nejaké obmedzenie?

Ďakujem

Odpovedať • Páči sa mi to



Anonymný Oct 17, 2021

Dobrý deň chcel som sa opýtať pri zadání sudoku máme overovať aj to či je sudoku zmysluplné ? to znamená či zadané sudoku neporušuje pravidla(ak sú 2 rovnaké čísla v jednom riadku)

Odpovedať • Páči sa mi to