

Anotace

Program, který hráči vygeneruje náhodné sudoku s jím vybraným počtem odhalených číslic a umožní mu je řešit.

Přesné zadání

Program, který umožní hráči vygenerovat sudoku s 26-81 odhalenými číslicemi, které je řešitelné (tj. má právě jedno řešení). Následně mu sudoku umožňuje řešit za použití pomůcek – vepisování a mazání číslic do buněk, vepisování a mazání poznámek a kontroly řešení.

Zvolený algoritmus

Generace sudoku funguje následovně. Začneme s prázdnou tabulkou, kterou následně celou vyplníme podle pravidel sudoku. Tím dostaneme validní řešení.

Tento proces provádíme rekurzí pomocí průchodu do hloubky, zkusíme vepsat do buňky náhodnou možnou hodnotu (možnou – tj. takovou, která splňuje pravidla sudoku), pokud žádná neexistuje, vrátíme se zpět k předchozí buňce, kde zkusíme jinou hodnotu.

Následně vezmeme vyplněnou tabulku a vymažeme z ní potřebný počet buněk. Odstraňování políček opět děláme rekurzí pomocí průchodu do hloubky, vybereme náhodnou buňku, kterou se pokusíme vymazat. Pokud výsledné sudoku má právě jedno řešení, tak se pokusíme smazat další políčko, pokud ne, pokusíme se smazat jinou buňku. Pokud už byla vyzkoušena všechna políčka, vrátíme se zpět a u předchozího kroku zkusíme jinou buňku.

Kontrola počtu řešení je provedena pomocí rekurzivního podprogramu, který v základu funguje na principu hrubé síly zkoušení všech řešení, ale pro optimalizaci je upravený.

Nejprve do všech políček, ve kterých je jen jedna možnost, tuto hodnotu zapíše. Když už zbývají jen políčka s dvěma více možnými hodnotami, program vezme políčko s nejméně možnostmi a postupně do něj zkusí všechny vepsat. Jakmile do něj nějakou hodnotu zapíše, funkce zavolá sama sebe a rekurzivně doplní zbytek tabulky. Pokud vyplní celou tabulku, zvýší se počet řešení o 1. Jakmile funkce najde 2. řešení, skončí a dává vědět, že pozice netvoří vhodné sudoku. Pokud nalezne pouze 1 řešení, vrátí True, pozice je validní sudoku.

Diskuze výběru algoritmu

Rozhodovala jsem se mezi dvěma hlavními přístupy.

První, který jsem zvolila, tj. nejprve celé sudoku vyplníme a poté odstraníme potřebný počet políček.

A druhý - do tabulky rovnou zkoušíme rekurzivně vepsat právě daný počet číslic.

Rozhodla jsem se pro první přístup, jelikož ač u prvního musíme provést více kroků, druhý přístup mi i přesto přišel mnohem pomalejší. U druhého přístupu totiž musíme nejen zkoušet všechny kombinace políček, která mohou být vyplněna, ale i hodnot v nich. Existuje tak velmi mnoho možností, z nichž jen málo bude vhodných.

Další rozhodnutí bylo nutné provést u koncepce algoritmu pro kontrolu řešení. Vystala otázka toho, co vlastně je řešitelné sudoku. Volba byla mezi kontrolou pomocí hrubé síly, kdy se kontrolují všechny možné vyplnění, a kontrolou pomocí lidských přístupů.

Při prvním přístupu je možné, že vzniknou i sudoku, kde jediná cesta vpřed je pomocí zkoušení a backtrackingu. To se může zdát nezábavné, a proto by se mohlo zdát dobré použít druhý přístup. Na druhou stranu první přístup zaručuje, že opravdu existuje jen jedno správné řešení, ať bude uživatel postupovat jakkoliv. Proto jsem se nakonec rozhodla pro první možnost.

Program

Program se dělí na dvě hlavní části:

Logickou část generování sudoku, kterou zařizuje funkce `vygeneruj_sudoku` a třída `policko`.

A grafiku, jež je obstaraná funkcí `hra`, která zajišťuje běh programu, menu a veškerou interakci s uživatelem. Pro pomoc s grafikou také existují třídy `pole_s_textem`, `tlacitko` a `prazdne_pole_sudoku`, které slouží k zobrazování různých prostředků s textem.

Třída `policko` slouží pro vytváření objektů jednotlivých políček sudoku. Každé políčko si udržuje informace o sobě jako hodnotu, kterou má v sobě vepsanou, hodnoty, které se do něj mohou vepsat, a své sousedy – políčka, která ho ovlivňují. Objekty třídy `policko` mají dvě hlavní metody `smaz_hodnotu` a `vepis_hodnotu`, které umožňují upravit hodnotu napsanou v políčku a zároveň dají vědět všem sousedům políčka o změně, aby si mohli upravit své seznamy možných hodnot. Tyto metody se volají, kdykoli při generování sudoku manipulujeme s hodnotou vepsanou v políčku.

Hlavní proměnná, kterou využíváme v celém programu, je proměnná `policka`, ve které je uložený seznam všech políček. Místo pak také pracujeme s proměnnými `prazdna_policka` a `vyplnena_policka`, pokud se nějaké akce týkají jen prázdných/plných políček.

Funkce `vygeneruj_sudoku` zařídí celé samotné generování, jako vstup bere počet číslic, které mají být v sudoku odhalené, jako výstup vrací seznam všech políček a seznam prázdných políček.

Ke generování využívá několik podfunkcí:

`generuj_prazdnou_sudoku` – funkce, která vytvoří poprvé seznam všech políček, nejprve prázdných. Každé políčko inicializuje a nastaví mu jeho sousedy.

`generuj_plnou_sudoku` – funkce, která prázdnou tabulku celou zaplní dle pravidel sudoku. Využívá k tomu rekurzivní podfunkci `vypln_pole`, která se pokusí vyplnit jedno pole náhodnou možnou hodnotou a poté se pokusí vyplnit pole další.

A rekurzivní funkce `odstran_pole`, která se pokusí vymazat hodnotu z jednoho náhodného pole a zkontroluje, zda výsledné sudoku má stále právě jedno řešení, což udělá pomocí další funkce `jedno_reseni` – funkce, která právě toto zkoumá, a to za pomoci rekurzivní podfunkce `unikatni_reseni`, která se pokouší doplnit rozdělané sudoku do plného všemi možnými způsoby a počítá každé řešení, na které narazí.

Celou funkci `vygeneruj_sudoku` pak volá funkce `hra` v momentě, kdy od uživatele dostane jako vstup počet číslic.

Pro účel kontroly tohoto počtu má funkce `hra` podfunkci `zkontroluj_pocet_policek`.

Ostatní podfunkce třídy `hra`, stejně jako třídy `pole_s_textem`, `tlacitko` a `prazdne_pole_sudoku` slouží k práci s grafikou a modulem `pygame`.

Hlavní jádro funkce `hra` spočívá ve `while` cyklu, který běží, dokud běží program a uživatel ho neukončil.

Hra se může nacházet ve třech hlavních fázích – hlavním menu, získávání počtu číslic či řešení. V každé fázi je uživatel držen dalším `while` cyklem, dokud z ní neodejde vypnutím programu, či zadáním vhodného vstupu.

Pravidla sudoku

Hráč musí umístit do každého řádku, sloupce a čtverce 3×3 (vyznačeného tlustší čarou) všechny číslice 1-9 právě jednou. Tzn. že v jednom řádku, sloupci ani čtverci se žádná číslice nesmí vyskytovat dvakrát a žádná tam nesmí chybět.

Reprezentace vstupních dat a jejich příprava, reprezentace výstupních dat a jejich interpretace

Při zapnutí programu se ukáže hlavní menu, ve kterém se nacházejí dvě tlačítka. Při kliknutí na tlačítko levým tlačítkem myši se provede akce, která je na tlačítku napsaná. Po kliknutí na tlačítko „Ukončit program“ se celý program vypne, po kliknutí na tlačítko „Nové sudoku“ se uživateli zobrazí nová obrazovka.

Tlačítko „Ukončit program“ se nachází na téměř každé obrazovce, kterou uživatel uvidí, a vždy mu umožní program vypnout.

Na nové obrazovce bude napsaná žádost o zadání čísla od 26 do 81. Zde není potřeba na nic klikat, číslo je třeba napsat na klávesnici pomocí číslic, vše, co uživatel napsal, se ukáže v okénku uprostřed obrazovky. Je možné umazat poslední napsaný znak pomocí klávesy `backspace`. Klávesou `enter` se vstup potvrdí.

Pokud vstup neodpovídal zadání – tj. nejednalo se o číslo napsané číslicemi od 26 do 81, na obrazovce se objeví zpráva, která uživatele informuje o důvodu, proč vstup není vyhovující. Tato zpráva zmizí po

kliknutí levým tlačítkem myši na tlačítko „OK“ nebo zmáčknutí klávesy enter. Poté se objeví opět předchozí obrazovka se žádostí o vstup.

Pokud byl zadáný vstup vhodný, uživateli se zobrazí obrazovka řešení.

Na této obrazovce se nachází dole tlačítka „Nové sudoku“ a „Ukončit program“, která fungují stejně jako na úvodní obrazovce. Dále se zde nacházejí tlačítka „Vepsat hodnotu“, „Vepsat možnosti“ a „Vymazat hodnotu“. Toto jsou tři hlavní módy, které řešení nabízí, a vždy je zapnutý právě jeden. Tlačítko, které odpovídá zapnutému módu, má žlutý okraj a text. Na začátku je to mód „Vepsat hodnotu“.

V levé části obrazovky se nachází samotné sudoku, číslice, které jsou v něm napsané černou barvou, tvoří zadání. Řešení, které vepíše uživatel, bude napsáno modře. Uživatel může vždy psát či mazat hodnoty z jednoho políčka – tzv. aktivního políčka, které si zvolí. Aktivním políčkem mohou být pouze ta, která byla na začátku prázdná – tj. nejsou v nich napsané černé číslice.

Aktivní políčko se zvolí kliknutím levým tlačítkem myši na danou oblast v sudoku. Aktivní políčko má žluté pozadí. Pokud hráč klikne na aktivní políčko znovu, políčko aktivním být přestane. Pokud je nějaké políčko aktivní a hráč klikne na jiné, původní políčko aktivním být přestane a nové políčko se stane aktivním.

V módu „Vepsat hodnotu“ je možné do aktivního políčka vepsat hodnotu – zde se píše hodnota, kterou si je hráč jistý. Zápis probíhá následovně: hráč musí mít zvolen mód „Vepsat hodnotu“ a nějaké aktivní políčko, poté zmáčkne číslici 1-9 na klávesnici – tato číslice se napíše do políčka. Pokud v políčku byly napsány nějaké poznámky (viz další odstavec), přestanou být touto akcí vidět.

V módu „Vepsat možnosti“ je možné si do políčka psát malé číslice, které mohou sloužit uživateli jako poznámky. Tuto možnost je možné využít například pro psaní hodnot, které se v daném políčku mohou nacházet. Zápis probíhá následovně: hráč musí mít zvolen mód „Vepsat možnosti“ a nějaké aktivní políčko, poté zmáčknutím číslice 1-9 na klávesnici se tato hodnota buďto do políčka vepíše, pokud tam ještě není, nebo se z něj smaže, pokud tam už byla. Mazání možností je možné pouze přes tento mód „Vepsat možnosti“. Mód „Vymazat hodnotu“ funguje pouze na mazání hodnoty vepsané pomocí módu „Vepsat hodnotu“.

Pokud je v políčku napsaná hodnota, poznámky nejsou vidět, ale v políčku zůstávají, jakmile se hodnota odstraní, opět se ukážou.

V módu „Vymazat hodnotu“ je možné z políčka odstranit hodnotu, která je v něm napsaná. Je k tomu potřeba mít zvolený mód „Vymazat hodnotu“ a nějaké aktivní políčko. Poté stisknutím klávesy backspace nebo delete se z políčka smaže hodnota, pokud v něm nějaká je napsaná.

Poslední tlačítko je tlačítko „Kontrola: OFF“. Kliknutím na něj se nápis změní na „Kontrola: ON“. Tento mód nesouvisí se třemi módy „Vepsat hodnotu“, „Vepsat možnosti“ a „Vymazat hodnotu“, je vždy buďto zapnutý nebo vypnutý. Pokud je na tlačítku napsáno „Kontrola: OFF“, je mód kontroly vypnutý a nic se neděje. Pokud je na tlačítku napsáno „Kontrola: ON“, je mód kontroly zapnutý. Při zapnuté kontrole je každá vepsaná hodnota (ne možnost, ale pouze hodnota napsaná pomocí módu „Vepsat hodnotu“) kontrolována, zda je správná – tj. ta, která tam má být v plně vyřešeném sudoku. Pokud je hodnota správná, nic se neděje. Pokud správná není, číslice bude přeškrtnutá červenou čarou.

Průběh práce

Můj původní plán byl dovolit všechny možné počty odhalených číslic - tj. 17-81^[1]. Ale narazila jsem na problém, že ne každé řešení lze upravit na sudoku se 17 číslicemi. To mě přivedlo k otázce, co dělat dál.

Zvažovala jsem tři hlavní možnosti:

Zaprvé - limitovat počet číslic, které si uživatel může přát. Pro to jsem se nakonec rozhodla, i kvůli tomu, že i u počtů, které sice vždy redukovat jdou, někdy trvá najít vhodné sudoku příliš dlouho.

Zadruhé - pokud z řešení nelze udělat vhodné sudoku, vygeneruje se nové vyplněné sudoku. To se bude opakovat tak dlouho, dokud se nepodaří vytvořit požadované sudoku. Tato možnost mi přišla velmi pomalá, jelikož pokud z řešení nelze vytvořit sudoku, musí se prověřit všechny možnosti, což trvá dlouho a s generováním každého dalšího sudoku by se čas ještě více prodloužil. Navíc zde není garance úspěchu, jelikož plné sudoku se generuje náhodně, a tedy existuje možnost, že by se vhodné sudoku jednoduše nikdy nevytvořilo.

Zatřetí - implementovat pro menší počty číslic druhý přístup popsáný v kapitole Diskuze výběru algoritmu – tj. zkoušet číslice rovnou náhodně rozmísťovat. Jak už je popsáno v této kapitole, očekávala jsem, že tento přístup bude velmi pomalý.

I druhý i třetí přístup jsem zkusila implementovat a potvrdila se mi jejich očekávaná pomalost. Proces byl tak pomalý, že se nedal využít pro program, a proto jsem se nakonec rozhodla pro první možnost.

Co nebylo doděláno

Jak už bylo zmíněno, program zatím nedovoluje generovat sudoku s 17-25 číslicemi. Zatím jsem nepřišla na žádný úplně jiný algoritmus generování sudoku, a proto zůstává program s omezením.

Zdroje

1 McGuire, Gary, Bastian Tugemann, and Gilles Civario. "There is no 16-clue Sudoku: Solving the Sudoku minimum number of clues problem via hitting set enumeration." *Experimental Mathematics* 23.2 (2014): 190-217.