Sudoku Solver

Szerző: OTT ALEX, Pitlik László

Intézmény: KJE

Előzmények:<https://miau.my-x.hu/miau2009/index.php3?x=e0&string=sudoku>

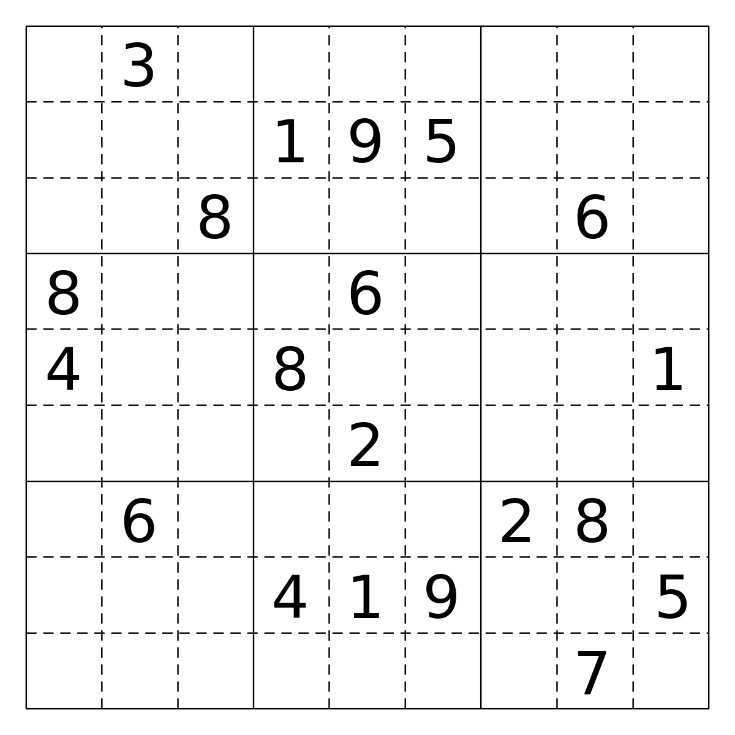
<https://miau.my-x.hu/miau2009/index.php3?x=e0&string=3333>

https://miau.my-x.hu/miau2009/index.php3?x=e0&string=csod%C3%A1k

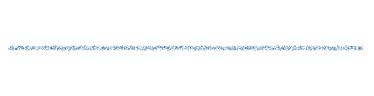
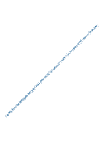
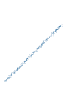
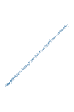
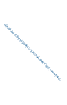
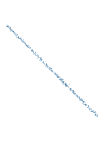
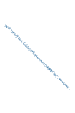
# Bevezetés

„A Sudoku egy logikai játék, melynek célja egy 9x9-es táblázat kitöltése számokkal úgy, hogy minden sorban, oszlopban és 3x3-as rész-táblázatokban[[1]](#footnote-1) (a továbbiakban „csempékben” is) az 1-től 9-ig terjedő számok mindegyike csak egyszer szerepeljen.

A játék elején a táblázat néhány mezője már ki van töltve, a többi mezőt pedig neked kell logikus gondolkodásod segítségével kitöltened. A Sudoku egy szórakoztató és kihívást jelentő játék, amely fejleszti a logikai készséget és a koncentrációt.” (Forrás: URL = ???)





A Sudoku megoldása nagyon sok képességet vesz igénybe, a logikai gondolkodást mondhatnánk a legfőbb szükségletnek, de ez közel sem elég, (hiszen akár térben[[2]](#footnote-2) is lehet gondolkodni).



A 9\*9-es Sudoku tehát 3x3 csempékből épül fel (vö. 1. ábra). 9 csempe alkot egy Sudoku-t. A megoldásához nagyon fontos a szabályok ismerete, ami több részből áll, de csak 1 egyszerű szabály hierarchikusan felépítve.



1. Számok 1-9 ig ismétlődnek.
2. Nem ismétlődhetnek a számok:
   1. átlóban? (vö. SUDOKU X[[3]](#footnote-3))
   2. sorban,
   3. oszlopban és a
   4. 3x3 csempéken belül sem
3. Ábra: cím, forrás

A Szabályok ennyire egyszerűek és pont ez teszi a játékot nehézzé hiszen a szabályt egy sms-ben is le lehet írni.

Itt kell értekezni arról röviden, hogy a SUDOKU nem triviálisan csak 2\*2, 3\*3, 4\*4, stb, vagyis vannak nem négyzetes kihívások is… Miért más ez?

Számomra a játék sokkal többet jelent, mint egy egyszerű „játék”. Gyerekkorom meghatározó időtöltése volt. A régi offline játékok, mint az Amőba, voltak könnyen elérhetőek, hiszen csak egy rácsos lapra volt hozzá szükség. Mind a kettő szabályzata hasonlóan egyszerű, könnyen fogyasztható és bárki által fél percen belül könnyen megtanulható.

A mesterséges intelligencia-alapú játékstratégia-felismerés kihívása azonban éppen az, hogy NEM szabad a szabályokat elmagyarázni = forráskódban, paraméterekben rögzíteni! Ez azért fontos, mert a Solver-alapú SUDOKU esetén egyelőre abból indul ki a szerző, hogy ezek a szabályok számára ismertek és így a Solver számára is megadásra kerülnek – kényszerűen egyelőre!

A Sudoku Solvert pont ez tette számomra hatalmas kihívássá és érdekes kihívássá, hiszen rengeteg időt töltöttem vele és most lehetőségem van nekem magamnak „tökéletesíteni”. A tökéletesítés itt és most azt jelenti, hogy a játékos spontán/intuitív szintjéről annak a lépésnek a kikerülésével kell egy automatikus megoldást jelentő matematikai keretet alkotni, hogy játékosként előtte meg kellett volna álmodni az univerzális = mindenkor sikert jelentő megoldási szabályrendszert (vö. szabályok ismeretének át NEM adása az MI számára). Vagyis tehát több rétege létezik a játékosi tudásnak:

1. Spontán találgatás (vö. quasi véletlenszerű próbálkozás = vak tyúk is talál szemet)
2. Intuitív ráérzés/logikus gondolkodás nem feltétlenül tudatosított komplex szabályrendszer, de tudatos részletszabályok alapján (okos találgatás = brute force a kritikus pontokon)
3. Tudatos, mindenkor egy előre (fejben, de akár papíron is) adott szabályrendszer követése (manuális robot)
4. A 3. szint szabályainak átírása forráskódba (automatizált szabály-elvű SUDOKU-megoldó szoftver)
5. Solver-alapú SUDOKU (= a szabályok beépítésre kerülnek a solver-logikába, DE a játékosnak NEM szükséges még a 2. szintet sem elérnie, vagyis nem kell tudnia megoldania egyedül, emberként a SUDOKU-t, pl. figyelem-zavaros személy esetén = definitív képtelenség, de a solver-paraméterezés kisebb koncentrációs szintű egyedi lépéseire még meg lehet a biológiai képesség)
6. MI-alapú automatizált felismerése a szabályok ismerete nélküli, csak a játékállapotokat és ezek értékét/értelmét jelentő inputoknak (vö. ha a játékállapotok ÉRTÉKE csak annyiban adott, hogy hibás lépés, ill. legitim lépés, ez már a szabályok indirekt becsorgatása a gépi tanulási folyamatba?!)
7. …

A feladat megoldása ennél viszont körökkel bonyolultabbnak bizonyult mert nem olyan egyszerű, mint azt bárki is hinné, hiszen egy számítógép csak arra képes, amit mi megadunk neki (vö. 1-6 lista fentebb az eltérő kihívásokról = az eltérő mit-is-adunk-meg stratégiákról). Kérdés: Vajon az MI jelenlegi tudása alapján nem képes önállóan megoldani egy SUDOKU-t emberi beavatkozás nélkül? Vagyis elérhető a 6. szint? De mire ez a dokumentum más által olvasásra kerül, lehet ez a kijelentés már elavulttá válik… A solver-alapúság valahol a normál szabály-elvűség és a klasszikus MI határán mozog:

* A solver pótolhatja a brute force-t
* A solver lehet genetikus/evolúciós algoritmussal támogatott
* A solver lehet egyenletrendszer-alapú, …

# Feladat-értelmezés

Először is meg kellett értenem a feladat lényegét = konvertálnom kellett a már eleve értett szabályokat a solver-paraméterezés (vö. speciális nyelv) szabályainak felhasználásával, vagyis el kellet magyaráznom a solver-paramétertér (nyelvi lehetőségek) segítségével mit jelent egy-egy szómágikus szabályrészlet solver-ül:

Több eshetőséget is meg kell vizsgáljunk, hogy meg tudjuk oldani. Megpróbáltam a „problémát” kisebb részletekre bontani, hogy könnyebben megérthető legyen, de az alap „probléma” a következő:

A red and blue squares with black numbers

Description automatically generatedSzükségünk van egy programra, ami megvizsgálja, hogy az általunk beírt számok előre meghatározott paramétereknek (szabályoknak?), hogyan felelhet meg? (megfelelnek-e egyáltalán?)

Jobbra lévő kép (2. ábra) egy általam írt próbálkozás a lehetséges végső megoldás irányába tett kísérletre.

Az Excel-képlet csak egyetlen dolgot vizsgál, ami nem más: mint,hogy az általunk megadott 9x9 es (nagy) csempéken mely számok (hányszor) fordulnak elő.

Hiszen egy-egy kockakövön (táblázati pozícióban, cellában) 1-9 ig fordulhat elő szám. Az adott 3x3-as (kis) csempén 9 számjegy fordulhat elő és mindegyik „unique” = nincs belőle még egy. A második általunk vizsgált szempont ,hogy az általunk beírt számok (1;…;9) sorban és oszlopban sem ismétlődnek, hiszen sorban és oszlopban is egyedi számokat kell tartalmaznia minden logikai egységnek (vö. nagy csempe, ill. játéktér).

A program kód itt csak azt vizsgálja,hogy a megadott 1x1 kockaköveken milyen egyedi számok jelennek meg , ahol kiveszünk egy számot ???? innentől nem értem a mondatot??? az összes lehetőség, ami ezt akadályozná azt kivesszük.

Nem világos még: mi a konklúziója a 2. ábrának?

1. Ábra: cím? Forrás? (pl. saját szerkesztés)

A grid of numbers with black text

Description automatically generatedEz volt az elsődleges ötletem,hogyan tudnám megvalósítani azt,hogy egy Sudoku megoldót készítsünk az Excel solverrel.

# Első próba

Először a „problémát” kisebb részre bontottam és a következő részletet a „mystyc 3x3” megoldó részletből, és a megoldáshoz vezető út már itt félre lett általam definiálva. Hiszen ez a program nem oldja meg az általam definiált „problémát” csak létrehoz (generál?) egy lehetséges végkifejletet. Itt viszont ezzel még nem voltam tisztában. Itt kellene kifejetni az alternatív megoldások számát:… pl. chatGPT-alapon?

A képlet (forrás: … xls, … munkalap, … tartomány, … solver-paramétersor = x. kép) itt még elég egyszerű volt. Adott egy 3 négyzet (cella) széles és 3 négyzet (cella) magas mágikus kocka, amiben egy szám egyszer szerepelhet 1től 9ig. A 15-ös szám a (kis) csempe szélén segített mert minden irányban meg kell lennie a 15nek ez azt eredményezi, hogy a csempe egyedi számokat fog használni, de ezzel erősen korlátozottak a megoldási lehetőségeink ~4db.<--FORRÁS? Levelezés?

# Második próba

Éreztem, hogy az első rész egy ígéretes irány, hiszen már sikerült egy-egy lehetséges megoldást generálnom. Ami egy ígéretes irány.

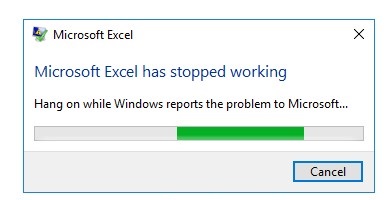
Az alternatív megoldások mindegyikét legenerálni pl. úgy lehet elkezdeni, hogy kikötjük, pl. a bal felső cella ne legyen az az érték, ami már az éppen adott sikeres megoldásban szerepel és újra futtatjuk a Solver-t.

A grid of numbers

Description automatically generatedMásodjára megpróbáltam🡨a teljes szövegre LEGYEN igaz = vagy én, vagy mi, vagy passzív!!!) a sikeresen generált 3x3 csempét 9 részre klónozni, hogy megoldást találhassak az esetleges 9x9 es Sudoku-ra. Itt a kihívás más megfogalmazásban az: hogyan lehet a fentebb előállítottalternatív (kis) csempéken egy nagy 9\*9-es helyesen kitöltött játéktérként értelmezni, lévén a nagy játéktér minden csempéje helyes kell, hogy legyen, s ezek vizuálisan ellenőrizhetően AKÁR ismétlődhetnek is, hacsak az nem egy spontán felismert új szabály, hogy legalább 9 alternatív kis (helyes) csempe létezik, s így ezek egy nagy 9\*9-es játéktéren SOHA nem ismétlődhetnek. Matematikusok számára szép kihívás annak bizonyítása, hogy létezik-e olyan SUDOKU egyáltalán, ahol minden kis csempe más, vagy nem minden kis csempe más? stb. (Ez nem a BPROF képzés szintje, de egy MSC (informatika?, ill. BSC matematika?) szint esetén már vélhetően ez a kihívás is kezelendő/kezelhető…)

x. ábra: cím, forrás

A betűtípus és betűméret a teljes szöveg testére LEGYEN azonos!

Sajnos ez a gondolat egyelőre még nem vezetett megoldásra, mert itt plusz 3 szabályt (lábjegyzetben mind a hármat le kell írni!) kellett bevezetni, aminek előzetesen még nem voltam tisztában, hogy sajnos ezen kinézete a táblázatnak ezt nem fogja engedni, mert az Excel Solver nem tud egymástól elhatárolt mezőket egybefüggően kezelni egy azonos szabályra, az összefűzés következténben pedig összeomlik (a kipróbált megoldás esetében legalább is). Sajnos az Excel log- vagy a windows log-ból nem sikerült azt kinyernem, hogy mi okozhatta a hibát, a Windowsnak írt hibaüzenetem a hiba konkrét leírására, pedig semmilyen válaszüzenetet nem eredményezett eddig. (vö. mellékletben képernyőképekkel alátámasztott kommunikáció)

Így a feladatban megpróbáltam a legtöbb tapasztalatot leszűrni intuitív módon,hogy mi lehet az, amivel tudnék fejldőni ebben a fázisban. (Na ez az az intuitivitás, amire egyelőre az MI a szó szoros értelmében nem képes: hiába látja be egy MI-alapú keretrendszer, hogy valami nem sikerült eddig, nem ijed meg, nem kapcsol túlélési fokozatba, nem álmodja meg a megoldást…)

Ami nem más lett, mint hogy a feladatot nem tagolhatom szét, (mert a Solver ezt közvetlenül nem tudja kezelni).

A következő részben? pedig a sor és oszlop nem lehet azonos számok. A képen (x. ábra) láthatjuk,hogy a megadott kritériumoknak megfelel, hiszen a sor, oszlop és átlók is stimmelnek a 15-ös összegszabályra és a 9x1 es kockaköves részek sorra és oszlopra is 45-öt adnak (45-ös összegszabály), ami nem más, mint 1-9 ig a számok összeadva.

x. ábra: cím, forrás

ITT kellene már mindenképpen az eddig felismert részletszabályok felsorolása (az mp4 alapján az ÖSSZES eddigi alternatívát egybe véve):

* 15 (indoklás:…)
* 45 (indoklás:…)
* … (indoklás:…)
* … (indoklás:…)
* … (indoklás:…)
* … (indoklás:…)
* … (indoklás:…)
* … (indoklás:…)
* … (indoklás:…)
* … (indoklás:…)

# A grid of numbers with a green box Description automatically generatedHarmadik próba

Előző 2 részből tanulva, már egybe a mozaik 9db 3x3-as csempe. 🡨magyartalan mondat?! Már az elején el kell dönteni és FEL IS KELL SOROLNI: milyen kulcsszavakat használ a szerző (táblázat, résztáblázat, csempe, kockakő, cella, pozíció, mátrix, mozaik) 🡨 csak egy minimális szókészlet a megengedett és nincs szinonima-használati JOG!

Oldalrésznek? (ez vajon mi?) minden 3x1 csempe 15 nek kell lennie🡨magyartalan?, ezt (mit?) a Solver nem tudta elvégezni, így csak 1x hajtotta vége a feladatot (mi volt a feladat) és a sorok oszlopok az ő értelmezésében a páros számból páratlan számot tudott generálni (hol lett bevezetve a páros és a páratlan előtte?), erre többedmagammal se találtam semmilyen magyarázatot, de a program alig pár ezred másodperc alatt erre a következtetésre jut (pontos solver-paraméter-kép, file, munkalap, tartomány, stb.). Így mivel elég sok kérdés maradt ezzel kapcsolatban bennem, úgy éreztem, hogy az Excel-nek nem adtam elég feltételt, hogy meg tudja oldani a problémát. Így ezt a próbálkozást tökéletesnek ítéltem meg arra vonatkozóan, hogy merre nem haladhatok tovább!

# Negyedik próba

Ezt a kísérletet érzem a legnagyobb tanulságomnak, mert az volt az a rész, ami a lehető leges-legmesszebb juttatott a valós Sudoku megoldás felé és ennek ellenére a legkomolyabb zsákutcámnak bizonyult.

A Kód (hol éri el az Olvasó?) nem alkalmas arra, hogy megoldjon Sudoku-t, de tökéletesen alkalmas arra, hogy egy-egy Sudoku-t létrehozzon:

Így egy fél sikernek éreztem, hiszen lehet nem vagyok képes megoldani A Problémát, de képes vagyok egy új tesztesetet létrehozni. A szabályok amit a programnak adtam egyszerűek voltak. Képernyőkép!!! (pl. az alább kövtkező még számozatlan ábra???)

Minden 3x3 AllDIfferent

Minden Sor AllDIfferent

Minden Oszlop AllDIfferent

Ez pedig a következő képet (x. ábra) eredményezte,hogy a sor számláló ami a különálló 3x3 csempéket adja össze 45 lesz, ezzel szemben a legelső próbához képest már közel se 15 az eredmény leggyakrabban előforduló száma.<--magyartalan, túl zavaros fogalmazás?

A screenshot of a crossword puzzle

Description automatically generatedJelen Sudoku, amit létrehoztam, közel se tudom, mennyi időbe telhetett mert 1 óra után és 1millió „subproblem” után én magára hagytam a gépet és lefeküdtem aludni.

x. ábra cím, forrás

De a program sikeresen tudott generálni Sudoku-t, tény és való, hogy ezzel szemben nem tud megoldani még egyet sem és abban az esetben, ha rögzítem a helyeket, hogy vegye bele a megoldásba akkor bizony elvérzik (= lefagy?). Mivel az AllDifferent kikötés miatt, nem tudok számokat rögzíteni?!

# Ötödik próba

A screenshot of a computer

Description automatically generatedEzen próba alkalmával a megoldást más irányból próbáltam megközelíteni, a Reddig (URL=…) segítségét vettem igénybe, ahol egy általam írt kódot tettem közzé és több ember segítségét kértem, hogy létrehozzunk egy VBS (rövidítés feloldása azonnal VBS = …) kódot az általam írt Python kódból. Ezt a kódot a végén egy már publikált szerző (név?) szolgáltatta számomra, hogy segítse a jelenlegi Sudoku Solver előre mozdulását a siker irányába, amivel a régi programot előre tudtam mozdítani mivel, sikerült egy VBA (rövidítés feloldása azonnal VBS = …) macro-t létrehoznom és ezzel az általam generált Sudoku-kat, amit generáltam, le tudom tesztelni, hogy helyesek-e? Sajnos ez a teszteset „csak” abban segített, hogy rájöjjek a „Negyedik próba” ban hol hibáztam és hogyan tudnám javítani. De ezzel a Solver nem mutatott sokkal nagyobb (hogyan mérjük?) javulást, de az idő, ami ahhoz kell, hogy sikerüljön megoldást találni, nagyban csökkent, már 1 óra után mikor elindítjuk(19:52 indítás első ellenőrzés 20:52) a generálást (Solver) akkor már az utolsó részeket tökéletesíti. Abban az esetben, ha sikerül a teljes folyamatot végig futtatni, természetesen dokumentálom tovább.

x. ábra cím, forrás

A screenshot of a computer

Description automatically generatedA két kép között további 1 óra telt el (22:02) és láthatólag a megoldás közelébe nem sikerül jutnia a Solvernak.x. ábta cím forrás 🡨minden ábrasorszámra a szöveg megfelelő pontján hivatkozni kell (ajánlatos az ábrákat a szöveg szerves részeként beilleszteni)

Felhasznált források:

<https://hu.wikipedia.org/wiki/Sz%C3%BAdoku>

<https://tutorialhorizon.com/excel/vba-excel-sudoku-solver/>

1. https://hu.wikipedia.org/w/index.php?title=Sz%C3%BAdoku&section=3# [↑](#footnote-ref-1)
2. https://hu.wikipedia.org/w/index.php?title=Sz%C3%BAdoku&section=3#Hagyom%C3%A1nyos\_sz%C3%BAdokuk [↑](#footnote-ref-2)
3. https://hu.wikipedia.org/w/index.php?title=Sz%C3%BAdoku&section=3#Hagyom%C3%A1nyos\_sz%C3%BAdokuk [↑](#footnote-ref-3)