## Korlátprogramozás bevezető

A feladatunkat úgynevezett korlátprogramozással oldottuk meg. Ennek a módszernek a legkorábbi verziója – a Sketchpad – 1963-ra tehető és [Ivan Sutherland](https://en.wikipedia.org/wiki/Ivan_Sutherland) nevéhez fűződik. Az 1980-as évektől egyre keresettebb lett, és mivel a logikai programozást szerették volna kiterjeszteni, így sok helyen korlát-logikai programozásként hivatkoznak rá. Az első praktikus verziókat – amiket üzleti célokra alkottak és már eladásra is bocsájtottak ­– az 1990-es években készítették. [[1]](#endnote-1)

A projektünkben a MiniZinc nevű grafikus szerkesztő programot használtuk – magát a nyelvet is így hívják –, aminek a fordítója az mzn2fzn, amely FlatZinc-re fordítja a MiniZinc modellt. Ezt a legtöbb megoldó által támogatott formátumú fájlt adja tovább a megoldónak, ami végül kiadja a megoldást. Az egyik legnépszerűbb megoldó a Gecode, melynek fő alkotója Christian Schulte. A munkát 2002-ben kezdték meg, 2005 decemberében adták ki az első verziót, és onnantól kezdve több évben is aranyérmes lett a kategóriájában. [[2]](#endnote-2) Ezen kívül több megoldóval is képes együttműködni a MiniZinc, ilyen például a Gurobi, a Chuffed, és a CBC.

A megoldó működését legkönnyebben egy feladaton keresztül lehet szemléltetni. A feladványt a szakirodalomban „négyszín tételként” szokták említeni. Adottak bizonyos országok és ezeket úgy kell kiszínezni adott számú színnel, hogy a szomszédos területek ne legyenek azonosak.

A modell elején megadjuk hány színt szeretnénk használni és külön megjegyezzük, hogy a megoldás során is ezeket vegyék fel az egységek. A szomszédsági mátrixot – ami egyben a feltételek listája is itt esetünkben – kikötésekben adjuk meg. A legprimitívebb mód erre az, hogy leírjuk páronként a szomszédokat, amiknek más és más értéket kell felvenniük, azaz két egymás mellett lévő rész nem lehet egyenlő tulajdonságú.



Térkép színezés

# MiniZinc modell

**%adatok**

**int: maxSzin = 4;**

**%megadjuk milyen értékeket vehetnek fel**

**var 1..maxSzin: Belgium;**

**var 1..maxSzin: Dánia;**

**var 1..maxSzin: Franciaország;**

MiniZinc logó

**var 1..maxSzin: Németország;**

**var 1..maxSzin: Hollandia;**

**var 1..maxSzin: Luxemburg;**

**%kikötések**

**constraint Belgium != Franciaország;**

**constraint Belgium != Németország;**

**constraint Belgium != Hollandia;**

**constraint Belgium != Luxemburg;**

**constraint Dánia != Németország;**

**constraint Franciaország != Németország;**

Gecode logó

**constraint Franciaország != Luxemburg;**

**constraint Németország != Hollandia;**

**constraint Németország != Luxemburg;**

**solve satisfy;**

A megoldó úgy dolgozik, hogy kiválaszt egy változót – a példában egy országot – és beállítja egy lehetséges értékre – színre. Ezután a korlátozások alapján következtetve csökkenti a többi változó lehetséges értékkészletét. Ezt a műveletet propagációnak nevezzük. A példában ez azt jelenti, hogy a szomszédos területeknél kizárja azt a lehetőséget, amit már felhasználtunk, így már csak a megmaradt színekből választhat.

Ha a propagáció során egy változó lehetséges értékkészlete üressé válik, akkor a megoldó visszavonja a legutolsó értékadást, és az adott értéket kizárja a változó értékkészletéből, mert ellentmondáshoz vezet. Ezután egy másik értéket ad neki, vagy egy új változót választ ki.

Amikor minden változónak sikerült értéket adni, akkor az egy lehetséges megoldása a feladatnak. Természetesen nem csak egy megoldás létezhet, ha például a piros és a fekete színt felcseréljük egymással, akkor az már másik megoldásnak fog számítani. Ha több megoldásra vagyunk kíváncsiak, folytathatjuk a keresést az utolsó értékadás visszavonásával. A korlátprogramozásnak ez egy hatalmas előnye más módszerekhez képest, hogy az összes lehetséges megoldást kiadja nekünk, nem pedig csak egyet.

A módszer optimalizálásra is használható, ahol a megoldások értékét egy célfüggvény adja meg. A keresés során egy további korlátozást kell figyelembe venni: hogy a megoldás értéke az eddig megtalált megoldásoknál jobb legyen.

1. [↑](#endnote-ref-1)
2. <http://www.gecode.org/presentations/Gecode%202011.pdf>

   <https://en.wikipedia.org/wiki/Gecode> [↑](#endnote-ref-2)