

A programról röviden

A program kétváltozós lineáris programozási feladat szemléltetését teszi lehetővé, grafikus módszerrel.

A feltételeknek megfelelően felépíti az S halmazt, majd a nívóvonal egy animáció során végighalad az extrémális pontokon, kiválasztva az optimálisat közülük.

Az animáció sebessége állítható. Az animáció megállítható/elindítható, és szükség esetén megismételhető.

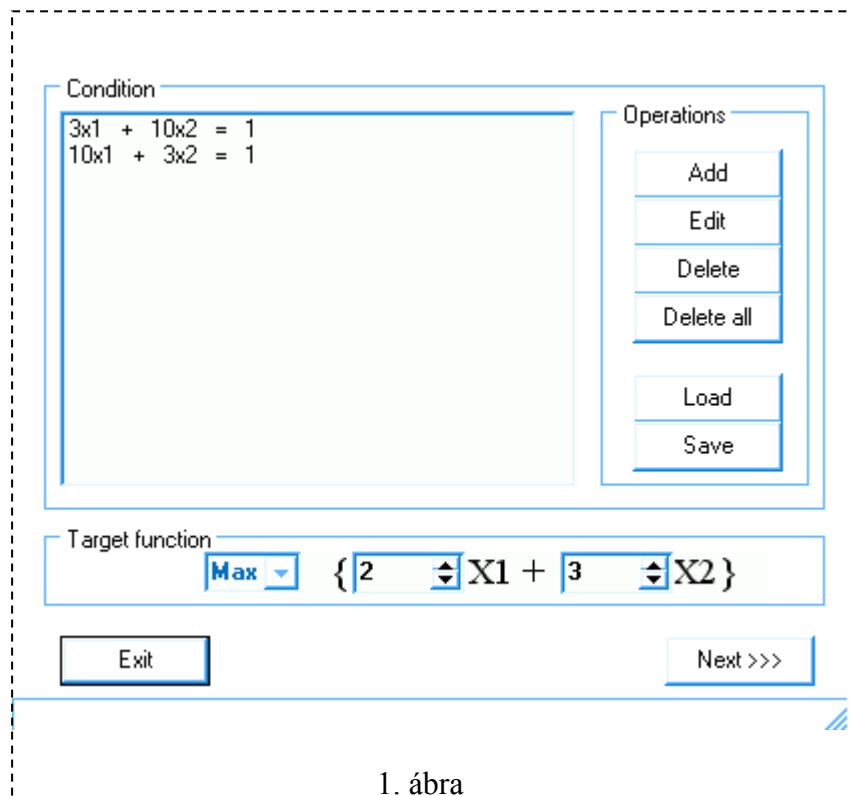
Az animáció végén megjelenik az eredmények ablak, amelyen fel van tüntetve az összes extrémális pont és az optimális megoldás.

A program használata

Az első ablakban lehet megadni a célfüggvényt, és a feltételeket.

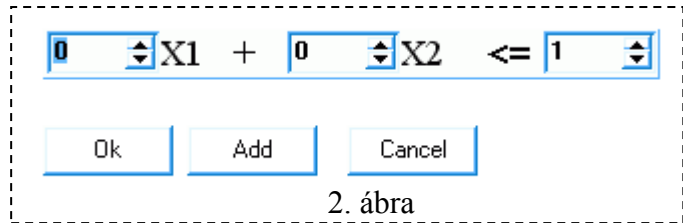
Műveletek:

1. ADD: új feltétel(ek) felvétele. (2. ábra)
2. EDIT: már meglévő feltételek módosítása.
3. DELETE: a kijelölt feltétel törlése.
4. DELETE ALL: az összes feltétel törlése.
5. LOAD: GraphGulf típusú feltételek betöltése.
6. SAVE: feltételek mentése (a célfüggvényt nem menti el)



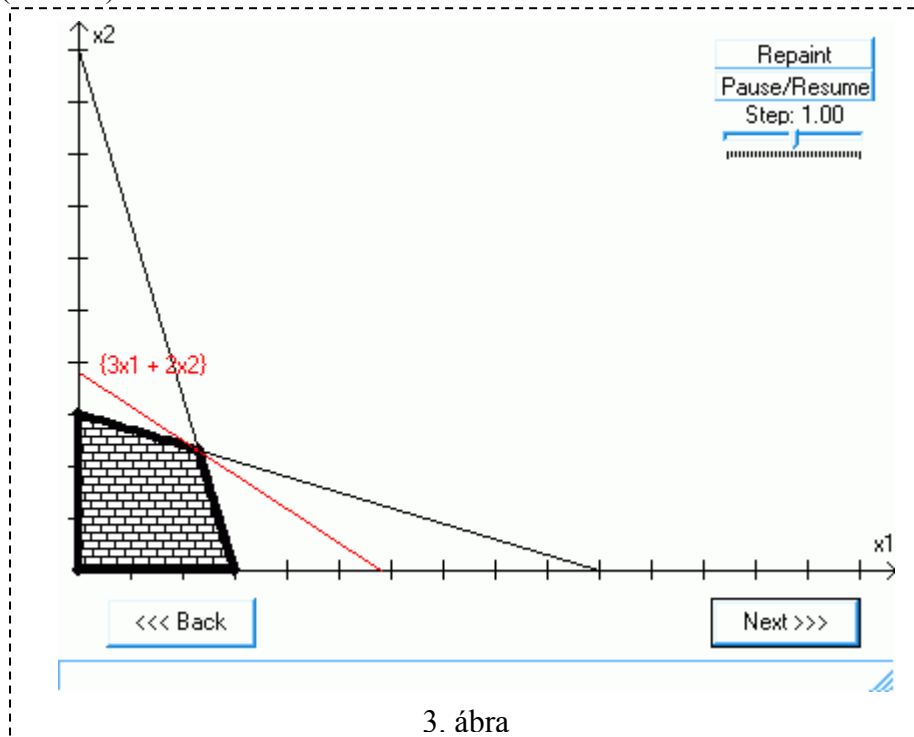
1. ábra

Az ADD gombra kattintva a 2. ablak nyílik meg, ahol lehetőségünk van változtatni az X1, X2, és a reláció értékén. (2. ábra)



2. ábra

A NEXT gombra kattintva (1. ábra), a program meghatározza az S halmazt és elindítja az animációt (3. ábra)

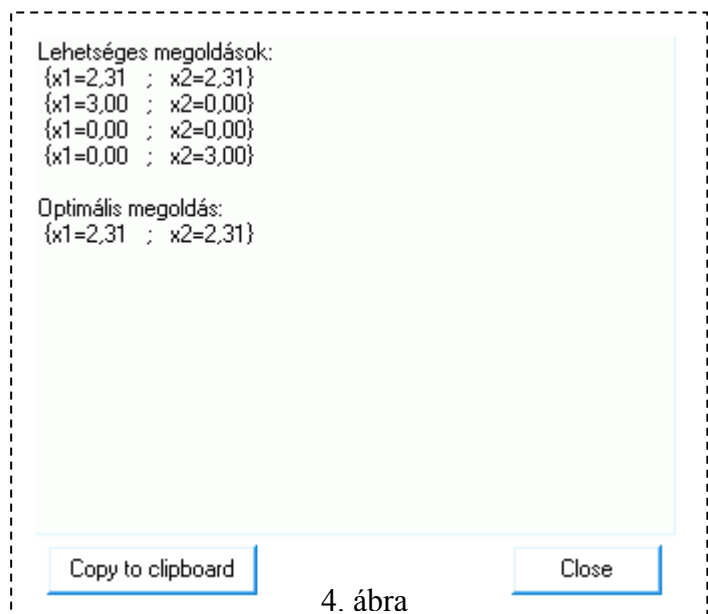


3. ábra

A REPAINT gombbal újra lehet indítani az animációt. A PAUSE/RESUME gombbal pedig pauzálni és folytatni lehet. A STEP a nívóvonal lépésközeit mutatja. Ha ez az érték nagy, akkor az animáció gyorsabban fog végrehajtódni.

Az animáció végéhez értével a 4. ábrán látható ablak fog megjelenni, az extrémális pontokkal és az optimális megoldásokkal.

Ezt a szöveget a COPY TO CLIPBOARD gombbal a vágólapra lehet helyezni.



4. ábra

A program, a programozó szemszögéből

Feltételek kezelése: A program a feltételeket egy dinamikus tömbben tárolja (Points). A tömb elemei egy-egy rekord, a rekord típusa pedig egyéni TGGPoint.

A TGGPoint definíciója:

```
Type TGGPoint = record  
  x1,x2   : integer;  
  relation : 0..5;  
  value   : integer;  
end;
```

A feltételeket először egy temp tömbbe helyezi (PointsTmp), így valósítható meg az hogyha az ok-ra kattintunk, akkor a feltételek hozzáadódnak a régebben felvett felvételekhez, ha viszont a cancel-re kattintunk akkor temp tömböt egyszerűen csak „elengedjük”.

A feltételek mentésekor egy TGGPoint típusú fájlba mentünk: var f : file of TGGPoint.

Az S halmaz felépítése: A program először az első feltételből felépít egy háromszöget, egy poligont (az S halmazt a továbbiakban csak poligonnak fogjuk emlegetni).

Majd sorról sorra veszi a feltételeket, és megnézi, hogy a poligon mely pontjai nem teljesülnek rá. Azokat kitörli. Mire az összes feltételt leellenőriztük a poligon pontosan egybe fog esni az S halmazzal.

A poligon csúcspontjai pedig az extrémális pontok.

Az animáció: A célfüggvényből meghatározunk egy egyenest. Majd ennek az egyenesnek mindkét komponenséhez STEP-nyit hozzáadunk, úgy hogy STEP megy 0-tól addig, amíg van metszéspontja a poligonnal. - A poligon és a nivóvonal metszését a program vastagabb vonallal rajzolja. –

Az optimális megoldás: Miután az animáció befejeződött a nivóvonal vagy pont metszi az optimális pontot, vagy túllépett rajta, mivel a STEP értéke elég nagy is lehet. Így az előző metszéspontot mindig el kell tárolni. Ez a metszéspont pontatlan lehet szintén a STEP miatt, ezért, meg kell vizsgálni, hogy melyik a poligon legközelebbi pontja az utolsó metszésponthoz képest és az lesz az optimális megoldás, hiszen az csak az extrémális pontok közül kerülhet ki. Ezt egy minimumkereséses algoritmussal és egy távolságméréssel lehet megoldani.