Operációkutatás I.

Szegedi Tudományegyetem
Informatikai Intézet
Számítógépes Optimalizálás Tanszék

3. Előadás

Szimplex algoritmus

Adott a standard alakú **LP feladat**:

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad i = 1,2,\ldots,m$$

$$x_j \geq 0 \qquad j = 1,2,\ldots,n$$

$$\max \sum_{j=1}^{n} c_i x_i = z$$

A standard LP feladathoz tartozó **szótár**:

$$x_{n+i} = b_i - \sum_{j=1}^{n} a_{ij} x_j \qquad i = 1,2,\ldots,m$$

$$z = \sum_{j=1}^{n} c_j x_j$$

Szimplex algoritmus

Most feltesszük, hogy $b_1 \geq 0, \ldots, b_m \geq 0$, azaz a feladathoz felírt szótár bázismegoldása

$$x = (0,0,\ldots,0,b_1,b_2,\ldots,b_m)$$

lehetséges megoldás.

- *A negatív jobb oldalú feltételt ($\exists i : b_i < 0$) tartalmazó standard feladatokkal később foglalkozunk*

A Szimplex algoritmus bevezetése

A szimplex algoritmus:

- Algoritmus: iteratív optimum keresés
- Ismételt áttérés más szótárakra a következő feltételek betartása mellett:
  1. Minden iteráció szótára ekvivalens az előző iterációéval
  2. Minden iteráció szótárának bázismegoldásán a célfüggvény értéke nagyobb vagy egyenlő, mint az előző iterációén
  3. Minden iteráció bázismegoldása lehetséges megoldás

Felmerülő kérdések:

- **Mi alapján térjünk át** egy másik szótárra?
- **Hogyan térjünk át**, hogy a feltételek teljesüljenek?
- **Honnan tudjuk**, hogy az aktuális bázismegoldás **optimális**?
- **Létezik-e** minden LP feladatnak **optimuma**?

Szimplex algoritmus - definíciók

Pivot lépés: új szótár megadása egy bázis és nembázis változó szerepének felcserélésével (ld. példa)

Belépőváltozó: a szimplex algoritmus egy iterációnak belépőváltozója az a nembázis változó, ami a következő szótárra áttérés hatására bázisváltozóvá válik

Kilépőváltozó: a szimplex algoritmus egy iterációnak kilépőváltozója az a bázisváltozó, ami a következő szótárra áttérés hatására nembázis változóvá válik

Szótárak ekvivalenciája: két szótár ekvivalens, ha az általuk leírt egyenletrendszer összes lehetséges megoldásai és a hozzájuk tartozó célfüggvényértékek rendre megegyeznek

Szimplex algoritmus

Tétel. *A pivot lépés előtti és az utána előálló új szótár ekvivalensek.*

Bizonyítás.

- Egyenleteket átrendezni ekvivalens átalakítás
- Egy egyenlethez egy másik konstans-szorosát hozzáadni ekvivalens átalakítás

∎

Szimplex algoritmus

Mi alapján térjünk át egy másik szótárra? Példa:

$$
\begin{array}{rcrcrcrcr}
x_4 & = & 4 & - & x_1 & - & 2x_2 & & \\
x_5 & = & 6 & - & x_1 & & & - & 4x_3 \\
x_6 & = & 2 & & & - & 2x_2 & + & 2x_3 \\
\hline
z & = & 10 & + & 3x_1 & + & 4x_2 & - & x_3
\end{array}
$$

- A célfüggvényértéke nő, ha az aktuális bázismegoldásban x_1 vagy x_2 értéke nő
- A célfüggvényértéke csökken, ha az aktuális bázismegoldásban x_3 értéke nő
- Melyik változó értékét növeljük, x_1 vagy x_2 értékét? Ld. **különböző pivot szabályok**.

Szimplex algoritmus

Hogyan térjünk át, hogy a feltételek teljesüljenek?

$$\begin{array}{rcrcrcrcr}
x_4 & = & 4 & - & x_1 & - & 2x_2 & & \\
x_5 & = & 6 & - & x_1 & & & - & 4x_3 \\
x_6 & = & 2 & & & - & 2x_2 & + & 2x_3 \\
\hline
z & = & 10 & + & 3x_1 & + & 4x_2 & - & x_3
\end{array}$$

- A bázisváltozók nemnegativitásából eredő korlátok x_2-re ($x_1 = 0, x_3 = 0$):

$$\begin{array}{ccccc}
x_4 \geq 0 & \Rightarrow & 4 - 2x_2 \geq 0 & \Leftrightarrow & 2 \geq x_2 \\
x_6 \geq 0 & \Rightarrow & 2 - 2x_2 \geq 0 & \Leftrightarrow & 1 \geq x_2
\end{array}$$

- A **legszűkebb korlátot** adó egyenletből fejezzük ki x_2-t (ld. hányadosszabály)

Szimplex algoritmus

Honnan tudjuk, hogy az aktuális bázismegoldás optimális?

$$
\begin{array}{rcrcrcrcr}
x_4 & = & 4 & - & x_1 & - & 2x_2 & & \\
x_5 & = & 6 & - & x_1 & & & - & 4x_3 \\
x_6 & = & 2 & & & - & 2x_2 & + & 2x_3 \\
\hline
z & = & & - & 3x_1 & - & 4x_2 & - & x_3
\end{array}
$$

Tétel. *Ha egy szótárban nincs pozitív c_j $(j = 1,2,\ldots,n+m)$ célfüggvény együttható és negatív b_i $(i = 1,2,\ldots,m)$ konstans a feltételek egyenleteiben, akkor a szótár bázismegoldása optimális megoldás.*

Bizonyítás. Indirekten könnyen belátható. ∎

Szimplex algoritmus

Létezik-e minden LP feladatnak optimuma?

$$
\begin{array}{rcrcrcrcr}
x_4 & = & 4 & + & x_1 & - & 2x_2 & & \\
x_5 & = & 6 & + & x_1 & & & - & 4x_3 \\
x_6 & = & 2 & & & - & 2x_2 & + & 2x_3 \\
\hline
z & = & & & 3x_1 & - & 4x_2 & - & x_3
\end{array}
$$

Látjuk, hogy x_1 tetszőlegesen növelhető az aktuális bázismegoldásban $\Rightarrow$ a célfüggvény így tetszőlegesen nagy értéket felvehet

Nem korlátos LP feladat: ha az LP feladat maximalizálandó (minimalizálandó) és célfüggvénye tetszőlegesen nagy (kicsi) értéket felvehet a lehetséges megoldásainak halmazán, akkor a feladatot **nem korlátosnak** nevezzük.

Szimplex algoritmus

$$
\begin{array}{ccccccccccc}
x_{n+1} & = & b_1 & - & a_{11}x_1 & - & a_{12}x_2 & \dots & - & a_{1n}x_n \\
x_{n+2} & = & b_2 & - & a_{21}x_1 & - & a_{22}x_2 & \dots & - & a_{2n}x_n \\
 & & & & & \vdots & & & & \\
\hline
z & = & z^* & + & c_1x_1 & + & c_2x_2 & \dots & + & c_nx_n
\end{array}
$$

Tétel. *Ha egy szótárban van olyan pozitív* c_j $(j = 1{,}2, \dots, n+m)$ *célfüggvény együttható, hogy minden* $-a_{ij}$ $(i = 1{,}2, \dots, m)$ *együttható nemnegatív, akkor az LP feladat, amihez a szótár tartozik, nem korlátos.*

Bizonyítás. Indirekten könnyen belátható.

■

A szimplex algoritmus

Bemenet/Input: Egy *lehetséges* induló szótár.

Kimenet/Output:

- $x^* = (x_1^*, \ldots, x_n^*)$ optimális bázismegoldás
- z^* a célfüggvény optimuma

(*Egy lehetséges induló szótárra van szükségünk, azaz feltesszük, hogy minden b_i konstans ($i = 1{,}2, \ldots, m$) nemnegatív az induló szótárban, azaz* **a szótár bázismegoldása lehetséges megoldás**. *Azzal az esettel, amikor ez nem teljesül, később foglalkozunk*)

A szimplex algoritmus

Az algoritmus lépései:

1. A szótárban $c_j \leq 0$ minden $j = 1,2,\ldots,n$-re?
   - Igen $\Rightarrow$ az aktuális bázismegoldás **optimális**, az algoritmus megáll
   - Nem $\Rightarrow$ folytatás a 2. ponttal
2. Válasszuk a nembázis változók közül belépőváltozónak valamely x_k-t, amelyre $c_k > 0$ (pozitív célfüggvény együttható)
3. $-a_{ik} \geq 0$ minden $i = 1,2,\ldots,m$-re?
   - Igen $\Rightarrow$ az LP feladat **nem korlátos**, az algoritmus megáll
   - Nem $\Rightarrow$ folytatás a 4. ponttal
4. Legyen l valamely index, amelyre $-a_{lk} < 0$ és $\left|\frac{b_l}{a_{lk}}\right|$ minimális (hányadosteszt)
5. Hajtsunk végre egy **pivot lépést** úgy, hogy x_k legyen a belépőváltozó és az l. feltétel bázisváltozója legyen a kilépő; folytatás az 1. ponttal

A szimplex algoritmus

Csak egy keretalgoritmus.

- Melyik változót válasszuk a belépőváltozónak a 2. pontban?
- Melyik feltételből válasszuk a kilépőváltozót a 4. pontban?

Pivot szabály: olyan szabály, ami egyértelművé teszi, hogy a szimplex algoritmusban mely változók legyenek a belépő- és a kilépőváltozók, ha több változó is teljesíti az alapfeltételeket

Klasszikus Szimplex algoritmus pivot szabálya:

- A **lehetséges belépőváltozók közül válasszuk a legnagyobb c_k értékűt**, több ilyen esetén azok közül a legkisebb indexűt
- **A lehetséges kilépőváltozók közül (hányadosteszt!) válasszuk a legkisebb l indexű** egyenlet változóját

Más pivot szabályok is léteznek. (ld. később)

Ciklizáció

Biztosan leáll az algoritmus véges számú lépés után?

- Mi van, ha az algoritmus első lépésében mindig nem a válasz (nem optimális az aktuális megoldás)?

Degenerált iterációs lépés: olyan szimplex iteráció, amelyben nem változik a bázismegoldás.

Degenerált bázismegoldás: olyan bázismegoldás, amelyben egy vagy több bázisváltozó értéke 0

Ciklizáció: példa

- *Példa*: degenerált iterációs lépés és bázismegoldás

- 1. iteráció

$$
\begin{array}{rcrrrrrrrr}
\boxed{x_5} & = & & - & \frac{1}{2}x_1 & + & \frac{11}{2}x_2 & + & \frac{5}{2}x_3 & - & 9x_4 \\
x_6 & = & & - & \frac{1}{2}x_1 & + & \frac{3}{2}x_2 & + & \frac{1}{2}x_3 & - & x_4 \\
x_7 & = & 1 & - & x_1 & & & & & & \\
\hline
z & = & & & \boxed{10x_1} & - & 57x_2 & - & 8x_3 & - & 24x_4
\end{array}
$$

- 2. iteráció

$$
\begin{array}{rcrrrrrrrrr}
x_1 & = & & & 11x_2 & + & 5x_3 & - & 18x_4 & - & 2x_5 \\
\boxed{x_6} & = & & - & 4x_2 & - & 2x_3 & + & 8x_4 & + & x_5 \\
x_7 & = & 1 & - & 11x_2 & - & 5x_3 & + & 18x_4 & + & x_5 \\
\hline
z & = & & & \boxed{53x_2} & + & 41x_3 & - & 204x_4 & - & 20x_5
\end{array}
$$

Ciklizáció: példa

- 3. iteráció

$$
\begin{array}{rcrrrrrrrr}
\boxed{x_1} & = & & - & \frac{1}{2}x_3 & + & 4x_4 & + & \frac{3}{4}x_5 & - & \frac{11}{4}x_6 \\
x_2 & = & & - & \frac{1}{2}x_3 & + & 2x_4 & + & \frac{1}{4}x_5 & - & \frac{1}{4}x_6 \\
x_7 & = & 1 & + & \frac{1}{2}x_3 & - & 4x_4 & - & \frac{3}{4}x_5 & - & \frac{11}{4}x_6 \\
\hline
z & = & & & \boxed{\frac{29}{2}x_3} & - & 98x_4 & - & \frac{27}{4}x_5 & - & \frac{53}{4}x_6
\end{array}
$$

- 4. iteráció

$$
\begin{array}{rcrrrrrrrr}
x_3 & = & & - & 2x_1 & + & 8x_4 & + & \frac{3}{2}x_5 & - & \frac{11}{2}x_6 \\
\boxed{x_2} & = & & & x_1 & - & 2x_4 & - & \frac{1}{2}x_5 & + & \frac{1}{2}x_6 \\
x_7 & = & 1 & - & x_1 & & & & & & \\
\hline
z & = & & - & 29x_1 & + & \boxed{18x_4} & + & 15x_5 & - & 93x_6
\end{array}
$$

Ciklizáció : példa

- 5. iteráció

$$
\begin{array}{rclllllll}
\boxed{x_3} & = & & & 2x_1 & - & 4x_2 & - & \frac{1}{2}x_5 & + & \frac{9}{2}x_6 \\
x_4 & = & & & \frac{1}{2}x_1 & - & \frac{1}{2}x_2 & - & \frac{1}{4}x_5 & + & \frac{5}{4}x_6 \\
x_7 & = & 1 & - & x_1 & & & & & & \\
\hline
z & = & & - & 20x_1 & - & 9x_2 & + & \boxed{\dfrac{21}{2}x_5} & - & \frac{141}{2}x_6
\end{array}
$$

- 6. iteráció

$$
\begin{array}{rclllllll}
x_5 & = & & & 4x_1 & - & 8x_2 & - & 2x_3 & + & 9x_6 \\
\boxed{x_4} & = & & - & \frac{1}{2}x_1 & - & \frac{3}{2}x_2 & + & \frac{1}{2}x_3 & - & x_6 \\
x_7 & = & 1 & - & x_1 & & & & & & \\
\hline
z & = & & & 22x_1 & - & 93x_2 & - & 21x_3 & + & \boxed{24x_6}
\end{array}
$$

Ciklizáció: példa

- 7. iteráció

$$
\begin{array}{ccccccccccc}
\boxed{x_5} & = & & - & \frac{1}{2}x_1 & + & \frac{11}{2}x_2 & + & \frac{5}{2}x_3 & - & 9x_4 \\
x_6 & = & & - & \frac{1}{2}x_1 & + & \frac{3}{2}x_2 & + & \frac{1}{2}x_3 & - & x_4 \\
x_7 & = & 1 & - & x_1 & & & & & & \\
\hline
z & = & & & \boxed{10x_1} & - & 57x_2 & - & 8x_3 & - & 24x_4
\end{array}
$$

- Visszakaptuk a kiindulási feladatot

Ciklizáció: ha a szimplex algoritmus valamely iterációja után **egy korábbi iteráció szótárát kapjuk meg újra**, akkor azt ciklizációnak nevezzük

Ciklizáció

Tétel. *Ha a szimplex algoritmus nem áll meg, akkor ciklizál.*

Bizonyítás. Tekintsünk két szótárat (melyeket a szimplex algoritmus során kapunk) amelyeknek ugyanazok a bázisváltozói:

- 1. szótár

$$
\begin{array}{ccccccccc}
x_4 & = & b_1 & - & a_{11}x_1 & - & a_{12}x_2 & - & a_{13}x_3 \\
x_5 & = & b_2 & - & a_{21}x_1 & - & a_{22}x_2 & - & a_{23}x_3 \\
x_6 & = & b_3 & - & a_{31}x_1 & - & a_{32}x_2 & - & a_{33}x_3 \\
\hline
z & = & v & + & c_1x_1 & + & c_2x_2 & + & c_3x_3
\end{array}
$$

- 2. szótár

$$
\begin{array}{ccccccccc}
x_4 & = & b'_1 & - & a'_{11}x_1 & - & a'_{12}x_2 & - & a'_{13}x_3 \\
x_5 & = & b'_2 & - & a'_{21}x_1 & - & a'_{22}x_2 & - & a'_{23}x_3 \\
x_6 & = & b'_3 & - & a'_{31}x_1 & - & a'_{32}x_2 & - & a'_{33}x_3 \\
\hline
z & = & v' & + & c'_1x_1 & + & c'_2x_2 & + & c'_3x_3
\end{array}
$$

Ciklizáció

- Ha egy $x_1, x_2, x_3, x_4, x_5, x_6, z$ megoldása az egyiknek, akkor a másiknak is (mivel a két szótár ekvivalens)
- Legyen x_k tetszőleges nembázis változó (pl. $x_k = x_1$), t tetszőleges valós szám
- Vegyük a következő megoldását az első szótárnak
  - $x_1 = t, x_2 = 0, x_3 = 0$
  - $\Rightarrow x_i = b_i - a_{i1}t \quad (i = 4{,}5{,}6)$ és $z = v + c_1 t$
- Ez megoldása a másik szótárnak is (mivel ekvivalensek), ezért:

$$b_i - a_{i1}t = b_i' - a_{i1}'t \qquad i = 4{,}5{,}6$$

$$v + c_k t = v' + c_k' t$$

Ciklizáció

Azaz a két szótár nem csak ekvivalens, hanem megegyezik.

- Általános esetben $\binom{n+m}{m}$ (véges!) különböző módon lehet bázist választani:

$$
\begin{array}{ll}
x_i = b_i - \sum_{j \in \mathcal{N}} a_{ij} x_j & \qquad i \in \mathcal{B} \\
\hline
z = v + \sum_{j \in \mathcal{B}} c_i x_j &
\end{array}
$$

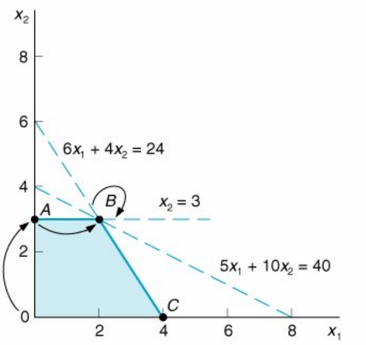
- **Ha az algoritmus nem áll meg**, akkor **lesz olyan iteráció**, amikor egy korábbi bázismegoldás, **korábbi szótár kerül elő újra**, azaz ciklizál.

■

A **ciklizáció elkerülhető megfelelő pivot szabály alkalmazásával**.

Illusztráció

Ciklizáció szemléletesen (2 változós feladat esetén): több korlátozó feltételhez tartozó egyenes megy át ugyanazon a ponton:



Bland szabály (1977)

Legkisebb index szabálya:

- A lehetséges belépőváltozók közül a legkisebb indexűt, ÉS
- A lehetséges kilépőváltozók közül a legkisebb indexűt választjuk

Tétel. *A szimplex algoritmus véget ér, ha a legkisebb index szabályt használjuk*

Bizonyítás. Nem bizonyítjuk. (*Egy bizonyítás megtalálható pl. Pluhár András: Operációkutatás I. jegyzetében*)

Lexikografikus szabály

A **ciklizáció oka a degeneráció**, azaz bázisváltozók 0 értékűvé válása a bázismegoldásokban. (*ld. Ciklizáció: példa*)

- Adjunk hozzá a kezdő szótár minden feltételének jobb oldalához egy pozitív ε_i konstanst (*mit jelent ez geometriailag?*)
- Minden feltételre legyenek ezek különböző nagyságrendűek a következőképpen
$$0 < \varepsilon_1 \ll \varepsilon_2 \ll \ldots \ll \varepsilon_m$$
- Szimplex algoritmus során biztosan nem "ütik ki" egymást
- Nem lesznek degenerált bázismegoldások $\Rightarrow$ nincs ciklizáció

Lexikografikus szabály

- *Példa*: feladat induló szótára (látjuk, hogy degenerált)

$$
\begin{array}{rcrrrrr}
x_3 & = & \frac{1}{2} & & & - & x_2 \\
x_4 & = & & - & 2x_1 & + & 4x_2 \\
x_5 & = & & & x_1 & - & 3x_2 \\
x_6 & = & & - & x_1 & + & 6x_2 \\
\hline
z & = & 4 & + & 2x_1 & - & x_2
\end{array}
$$

- Szótár konkrét értékek helyett szimbolikus ε-okkal kiegészítve

$$
\begin{array}{rcrrrrrrrrrrrrr}
x_3 & = & \frac{1}{2} & + & \varepsilon_1 & & & & & & & & & - & x_2 \\
x_4 & = & & & & + & \varepsilon_2 & & & & & - & x_1 & + & x_2 \\
x_5 & = & & & & & & + & \varepsilon_3 & & & + & x_1 & - & 3x_2 \\
x_6 & = & & & & & & & & + & \varepsilon_4 & - & 2x_1 & + & 4x_2 \\
\hline
z & = & 4 & & & & & & & & & + & 2x_1 & - & x_2
\end{array}
$$

Lexikografikus szabály

- 1. iteráció után

$$
\begin{array}{rcl}
x_3 &=& \frac{1}{2} + \varepsilon_1 - x_2 \\
x_4 &=& \varepsilon_2 - \frac{1}{2}\varepsilon_4 + \frac{1}{2}x_6 - x_2 \\
x_5 &=& \varepsilon_3 + \frac{1}{2}\varepsilon_4 - \frac{1}{2}x_6 - x_2 \\
x_1 &=& \frac{1}{2}\varepsilon_4 - \frac{1}{2}x_6 + 2x_2 \\
\hline
z &=& 4 + \varepsilon_4 - x_6 + 3x_2
\end{array}
$$

- 2. iteráció után

$$
\begin{array}{rcl}
x_3 &=& \frac{1}{2} + \varepsilon_1 - \varepsilon_3 - \frac{1}{2}\varepsilon_4 + \frac{1}{2}x_6 + x_5 \\
x_4 &=& \varepsilon_2 - \varepsilon_3 - \varepsilon_4 + x_6 + x_5 \\
x_2 &=& \varepsilon_3 + \frac{1}{2}\varepsilon_4 - \frac{1}{2}x_6 - x_5 \\
x_1 &=& 2\varepsilon_3 + \frac{3}{2}\varepsilon_4 - \frac{3}{2}x_6 - 2x_5 \\
\hline
z &=& 4 + 3\varepsilon_3 + \frac{5}{2}\varepsilon_4 - \frac{5}{2}x_6 - 3x_5
\end{array}
$$

Lexikografikus szabály

- Az utolsó szótár optimális
- Elhagyjuk az ε-okat:

$$
\begin{array}{rcrcrcr}
x_3 & = & \frac{1}{2} & + & \frac{1}{2}x_6 & + & x_5 \\
x_4 & = & & & x_6 & + & x_5 \\
x_2 & = & & - & \frac{1}{2}x_6 & - & x_5 \\
x_1 & = & & - & \frac{3}{2}x_6 & - & 2x_5 \\
\hline
z & = & 4 & - & \frac{5}{2}x_6 & - & 3x_5
\end{array}
$$

Lexikografikus szabály

Lexikografikus szabály:

- egészítsük ki szimbolikus ε konstansokkal az induló szótárat, majd
  - A **lehetséges belépőváltozók közül** válasszuk a **legnagyobb** c_k **értékűt**, több ilyen esetén azok közül a legkisebb indexűt
  - A **lehetséges kilépőváltozók közül** válasszuk azt, amelynek l indexű egyenletére az együtthatókból álló vektor **lexikografikusan a legkisebb**

Tétel. *A szimplex algoritmus biztosan véget ér, ha a lexikografikus szabály szerint választunk kilépő változót*

Bizonyítás. Nem bizonyítjuk.

Összefoglalva:

- A *legnagyobb növekmény* (ld. gyakorlat) és a klasszikus pivot szabály ciklizálhat
- A Bland és a lexikografikus szabály esetén nincs ciklizáció
- A *véletlen pivot* szabály választással az algoritmus 1 valószínűséggel terminál

Lexikografikus szabály

- *Példa*: a különböző pivot szabályok közötti különbségekre

$$
\begin{array}{rcrcrcrcrcr}
x_9 & = & 5 & - & 10x_1 & - & 5x_3 & - & x_5 & - & 3x_7 \\
x_8 & = & 1 & + & 5x_1 & + & 9x_3 & + & 21x_5 & - & 4x_7 \\
x_2 & = & 2 & + & 4x_1 & - & 2x_3 & - & 5x_5 & - & 8x_7 \\
x_4 & = & 4 & - & 8x_1 & + & 4x_3 & - & 5x_5 & - & 2x_7 \\
x_6 & = & 3 & - & x_1 & - & 2x_3 & - & 5x_5 & & \\
\hline
z & = & 27 & + & x_1 & + & 2x_3 & - & 11x_5 & + & 6x_7
\end{array}
$$

- A klasszikus, lexikografikus, legkisebb index és legnagyobb növekmény szabályok mind különböző belépő- kilépőváltozó párokat választanak (HF: ellenőrizzük!)

Boomington sörfőzde

A Bloomington Sörfőzde pilzenit és angol világos sört állít elő.

A pilzeni eladási ára 5$ hordónként, a angol világosé 2$ hordónként. Egy hordó pilzeni előállításához 5 kg árpamaláta és 2 kg komló szükséges. Egy hordó világos sörhöz pedig 2 kg árpamaláta és 1 kg komló kell. Rendelkezésre áll 60 kg árpamaláta és 25 kg komló.

Fogalmazzunk meg egy LP-t, amellyel maximalizálható a bevétel! Oldjuk is meg. Nézzük meg grafikusan is!

Reklámköltség minimalizálása

A Dorian Auto cég luxusautókat és teherautókat gyárt. A vállalat úgy gondolja, hogy vásárlói legnagyobb valószínűséggel magas jövedelmű nők és férfiak. Ennek a fogyasztói csoportnak a megnyerésére a cég egy komoly tévé-hirdetési kampányt indított és elhatározta, hogy 1 perces reklámhelyeket vásárol kétféle típusú tévéműsorban: vidám műsorokban és futballmeccsek alatt.

Minden kabarébeli reklámot 7 millió magas jövedelmű nő és 2 millió magas jövedelmű férfi néz. Minden futballmeccs alatti reklámot 2 millió magas jövedelmű nő és 12 millió magas jövedelmű férfi néz. Az egyperces kabarébeli reklám 50 ezer dollárba kerül, és az egyperces futballmeccs alatti reklám ára 100 ezer dollár. A cég azt szeretné, ha hirdetéseit legalább 28 millió magas jövedelmű nő és 24 millió magas jövedelmű férfi látná.

Alkalmazzuk a lineáris programozást arra, hogy a Dorian cég a reklámcéljait minimális költségek mellett érje el!