

VELEUČILIŠTE U POŽEGI



MONIKA KNEŽEVIĆ

**PRIMJENA LINEARNOGA PROGRAMIRANJA U
PLANIRANJU PROIZVODNJE**

DIPLOMSKI RAD

Požega, 2013. godine.

VELEUČILIŠTE U POŽEGI
DRUŠTVENI ODJEL
SPECIJALISTIČKI DIPLOMSKI STRUČNI STUDIJ
TRGOVINSKO POSLOVANJE

**PRIMJENA LINEARNOGA PROGRAMIRANJA U
PLANIRANJU PROIZVODNJE**

DIPLOMSKI RAD

PREDMET: KVANTITATIVNE METODE U
TRGOVINI

MENTOR: mr.sc. Bojan Kovačić

STUDENT: Monika Knežević

Požega, rujan 2013. godine.

SADRŽAJ

1. UVOD	1
2. OSNOVE LINEARNOGA PROGRAMIRANJA	2
2.1. Kratka povijest linearnoga programiranja	2
2.2. Osnovna načela linearnoga programiranja	2
3. RAČUNALNI PROGRAM WINQSB	6
3.1. Modul za linearno i cjelobrojno linearno programiranje <i>Linear and Integer Programming</i>	6
4. PRIMJENA LINEARNOGA PROGRAMIRANJA NA PROBLEME PLANIRANJA PROIZVODNJE	12
4.1. Primjer 1	12
4.2. Primjer 2	17
4.3. Primjer 3	19
4.4. Primjer 4	25
4.5. Primjer 5	28
4.6. Primjer 6	33
4.7. Primjer 7	35
4.8. Primjer 8	37
5. ZAKLJUČAK.....	43
6. POPIS KRATICA I AKRONIMA	44
7. POPIS LITERATURE I DRUGIH IZVORA PODATAKA.....	45

1. UVOD

Cilj ovog rada je pokazati jednostavnost primjene računalnoga programa WinQSB u rješavanju različitih problema linearnoga programiranja vezanih uz proces proizvodnje. Posebno će se razmatrati analiza osjetljivosti optimalnoga rješenja.

Računalni program za linearno programiranje može pomoći mnogim ljudima: od npr. domaćice koja želi ispeći što više kolača za svatove od različitih sastojaka koje posjeduje do npr. velikog proizvođača namještaja koji želi odrediti kako iskoristiti ograničene količine sirovina uz najveći profit.

U drugom poglavlju bit će opisana kratka povijest razvoja linearnoga programiranja koje je relativno nova grana matematike, te njegova osnovna načela.

U trećem poglavlju objasnit će se korištenje modula za linearno i cjelobrojno linearno programiranje u programu WinQSB.

U četvrtom poglavlju prikazat će se detaljno rješavanje odabranih primjera linearnoga programiranja iz područja trgovinskoga poslovanja, a uz korištenje programa WinQSB. Promatrat će se maksimiziranje korisnosti ili minimiziranje troškova uz zadane uvjete. Izradit će se i analiza osjetljivosti optimalnoga rješenja, odnosno utvrdit će se što se dogodi s optimalnim rješenjem ako dodamo neki novi uvjet ili malo promijenimo neku vrijednost u pripadnom matematičkom modelu.

2. OSNOVE LINEARNOGA PROGRAMIRANJA

Linearno programiranje je namijenjeno rješavanju problema optimalne linearne kombinacije više nezavisnih varijabli radi ostvarenja maksimalnog ili minimalnog rezultata uz zadovoljenje postavljenih uvjeta i/ili zahtjeva. Optimizacija uključuje odabir varijabli i određivanje njihovih vrijednosti. U nastavku će se navesti kratka povijest linearnoga programiranja i njegova osnovna načela.

2.1. Kratka povijest linearnoga programiranja

Linearno programiranje je grana matematike koja se bavi problemom optimizacije sustava unutar zadanih uvjeta. U povijesti razvoja linearnoga programiranja posebna uloga pripisuje se nekolicini svjetski poznatih matematičara. Ruski matematičar Leonid Kantorovich prvi put je 1939. godine uveo pojam linearnoga programiranja u rješavanju problema optimalne potrošnje resursa.[6]

U SAD-u je linearno programiranje razvijeno tijekom drugog svjetskog rata prvenstveno za probleme vojne logistike, kao što je optimiziranje prijevoza vojske i opreme konvojima. Važan je i doprinos ekonomista Tjallinga Koopmansa. Kantorovic i Koopmans su 1975. god. podijelili Nobelovu nagradu za ekonomiju za svoj pionirski rad u linearnom programiranju.[9]

1947.godine Georg Dantzig objavljuje simplex metodu za rješavanje problema linearnoga programiranja. Iste godine John von Neumann razvija teoriju dualnosti. 1979. godine Leonid Khachiyan objavljuje dokaz o rješivosti problema linearnoga programiranja u polinomijalnom vremenu, dok je širu teorijsku i praktičnu pozadinu dao Narendra Krishna Karmarkar 1984. godine uvodeći metodu unutarnje točke.[6]

2.2. Osnovna načela linearnoga programiranja

Tipični problemi linearnoga programiranja su npr:

- izraditi plan korištenja ograničene količine sirovina uz najveći profit;
- izraditi raspored zaduženja djelatnika koji objavljuju neki posao tako da taj posao bude napravljen u najkraćem mogućem vremenskom roku.

Cilj ovih problema je optimizacija, odnosno maksimiziranje korisnosti ili minimiziranje troškova uz zadane uvjete što se rješava linearnim programiranjem. Područje primjene linearnoga programiranja je široko: proizvodnja, transport i distribucija, marketing, telekomunikacije, financijsko ulaganje i planiranje, izrada rasporeda zaposlenika i sl. Formulirati (modelirati) realni životni problem kao problem linearnoga programiranja zahtijeva timski rad stručnjaka iz više područja.[9]

U ekonomskim problemima neka ekonomska veličina (varijabla) često zavisi o najmanje dvije druge ekonomske veličine (varijable) Tako npr. količina potražnje q nekoga dobra ne ovisi samo o cijeni p_1 toga dobra, nego i o cijeni p_2 nekoga drugog dobra (komplementarnoga ili supstituta). To onda znači da vezu između tih veličina (varijabli) ne možemo iskazati funkcijom jedne varijable, nego moramo promatrati funkcije dviju ili više varijabli. U ekonomiji se često pojavljuju i problemi koje je moguće formulirati u obliku matematičkoga modela s funkcijom cilja koju treba maksimizirati ili minimizirati i ograničenjima u obliku jednadžbe. Metodama rješavanja takvih problema bavi se matematičko programiranje. [2]

Matematičko programiranje načelno obuhvaća linearno i nelinearno programiranje. Ono zapravo predstavlja poopćenje klasičnih optimizacijskih modela. Klasični optimizacijski model obuhvaća razmatranje funkcije cilja uz uvjete iskazane u obliku jednadžbi ili sustava jednadžbi. U matematičkom se programiranju razmatraju i matematički modeli u kojima su uvjeti zadani u obliku nejednadžbi. Na primjer, umjesto zahtjeva da potrošač potroši točno 250 kn, okvir matematičkoga programiranja omogućava slobodu izbora utroška, pa utrošak može iznositi ili barem 250 kn ili najviše 250 kn. „Liberalizacijom“ uvjeta ovaj strukturni okvir optimizacije odmah čini problem zanimljivijim i realističnijim. Međutim, on također inicira razvijanje novih metoda rješavanja jer se za uvjete u obliku nejednadžbi ne mogu relativno jednostavno upotrijebiti uobičajene metode diferencijalnoga i integralnoga računa.

Linearno programiranje je jednostavnija varijanta matematičkoga programiranja u kojoj su funkcija cilja i uvjeti u obliku nejednadžbi funkcije oblika

$$f(x_1, \dots, x_n) = \sum_{i=1}^n a_i \cdot x_i.$$

Takve funkcije uobičajeno se nazivaju linearne funkcije, pa odatle i naziv za ovu vrstu matematičkoga programiranja. [1]

U slučaju dviju varijabli ($n = 2$) optimalno rješenje problema linearnoga programiranja vrlo jednostavno možemo dobiti grafički (grafičkom metodom) neovisno o ukupnom broju uvjeta u pripadnom matematičkom modelu. Naime, dodatni uvjeti mogu samo povećati ukupan broj ekstremnih točaka, ali ne i dimenziju n . Međutim, rješava li se matematički model s $n = 3$ varijable odlučivanja, grafička metoda postaje relativno komplicirana jer će se zahtijevati trodimenzionalni grafički prikaz. Za rješavanje bilo kojega matematičkoga modela u kojemu se pojavljuju najmanje četiri varijable odlučivanja ($n = 4$) grafička metoda je potpuno neprimjenjiva. Prema tome, nužno je tražiti negrafičku metodu rješavanja koja se može primijeniti na rješavanje matematičkoga modela s proizvoljnim brojem nezavisnih varijabli.[1]

Mnogi problemi koji se javljaju kako u praksi, tako i u znanosti, prevedeni na matematički oblik, sastoje se od određivanja ekstrema (maksimuma i minimuma) neke funkcije uz zadane uvjete. Pritom funkcija čiji se ekstrem želi odrediti obično ima određeno značenje, kao što su npr. profit, troškovi itd., dok uvjeti, najčešće u obliku nejednadžbi ili jednadžbi, predstavljaju tehnološke, tržišne i druge uvjete za zadani problem. [3]

Optimalno rješenje uvijek je jedna točka iz skupa mogućih rješenja. No, za njezino određivanje ne treba ispitivati sve elemente toga skupa. Taj je skup najčešće neki ravninski lik, geometrijsko tijelo, odnosno konveksni simpleks u općem slučaju. Tada je optimalno rješenje u pravilu jedan vrh takvoga matematičkoga objekta iako treba napomenuti da postoje slučajevi u kojima je skup optimalnih rješenja npr. stranica konveksnoga n -terokuta, brid ili jedna ploha geometrijskoga tijela itd. Takvi slučajevi u ovome radu neće biti razmatrani.

Taj rezultat zasniva se na činjenici da je, bez obzira na broj varijabli odlučivanja, skup mogućih rješenja linearnoga programiranja uvijek zatvoren konveksni skup. [1] Podsjetimo, skup $S \subseteq R^n$ je otvoren ako oko svakoga elementa skupa S možemo opisati kružnicu ili kuglu K koja pripada skupu S , a zatvoren ako je skup $R^n \setminus S$ otvoren. Sve „klasične“ geometrijske figure poput kružnice, trokuta, pravokutnika, kvadra itd. su zatvoreni skupovi. Napokon, skup S je konveksan ako za bilo koje dvije njegove točke A i B spojnica \overline{AB} također pripada skupu S . [3]

Standardni oblik problema linearnoga programiranja je:

$$\max z = f(x_1, \dots, x_n) = \sum_{i=1}^n c_i \cdot x_i$$

pod uvjetima

$$a_{11} \cdot x_1 + \dots + a_{1n} \cdot x_n \leq b_1$$

...

$$a_{m1} \cdot x_1 + \dots + a_{mn} \cdot x_n \leq b_m$$

Pritom se funkcija z uobičajeno naziva funkcija cilja, a varijable x_1, \dots, x_n varijable odlučivanja. Nerijetko se postavljaju dodatni zahtjevi da vrijednosti tih varijabli budu nenegativni realni brojevi, prirodni brojevi i sl.

Neka područja primjene linearnoga programiranja navedena su u donjoj tablici.

Tablica 3.1. Područje primjene linearnoga programiranja [8]

Područje primjene	Primjer
Financije	<ul style="list-style-type: none"> • Investiranje (maksimiziranje dobiti) • Investiranje (minimiziranje rizika)
Marketing	<ul style="list-style-type: none"> • Distribucija oglašavanja (marketinški mix)
Proizvodnja	<ul style="list-style-type: none"> • Maksimalna dobit • Minimalni troškovi • Optimalna distribucija proizvoda
Nabava	Maksimalna dobit
Naftna industrija	<ul style="list-style-type: none"> • Optimalna mješavina nafte
Prehrambena industrija	<ul style="list-style-type: none"> • Mješavina sastojaka za konačni proizvod • Maksimiziranje dobiti od product mix-a
Izgradnja pogona	<ul style="list-style-type: none"> • Traženje optimalne pozicije za izgradnju pogona gdje će troškovi prijevoza biti minimalni

3. RAČUNALNI PROGRAM WINQSB

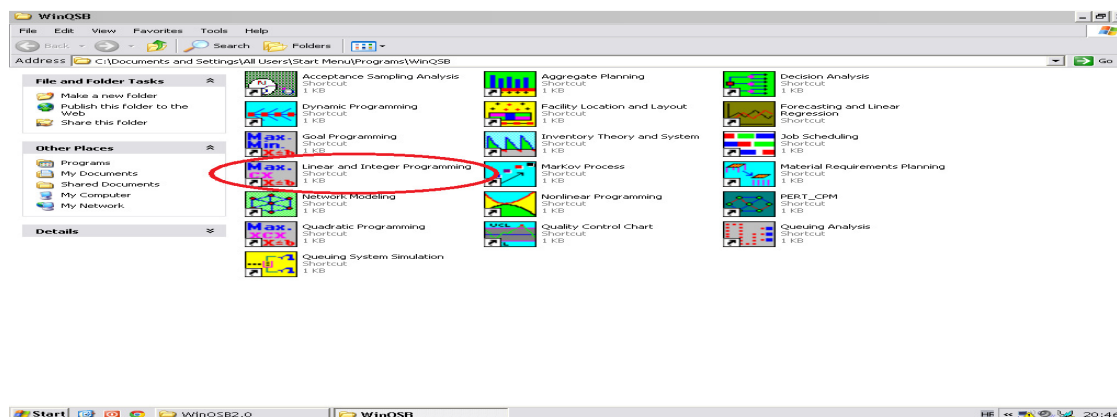
WinQSB je kvantitativni sustav za poslovanje napravljen za Windows sučelje. Sastoji se od ukupno devetnaest aplikacijskih modula. U nastavku će biti opisano korištenje modula za linearno i cjelobrojno linearno programiranje koje će biti korišteno u rješavanju odabranih primjera iz područja proizvodnje.

3.1. Modul za linearno i cjelobrojno linearno programiranje *Linear and Integer Programming*

Linear and Integer Programming je programski modul koji rješava probleme linearnoga programiranja i cjelobrojnoga linearnoga programiranja. [7]

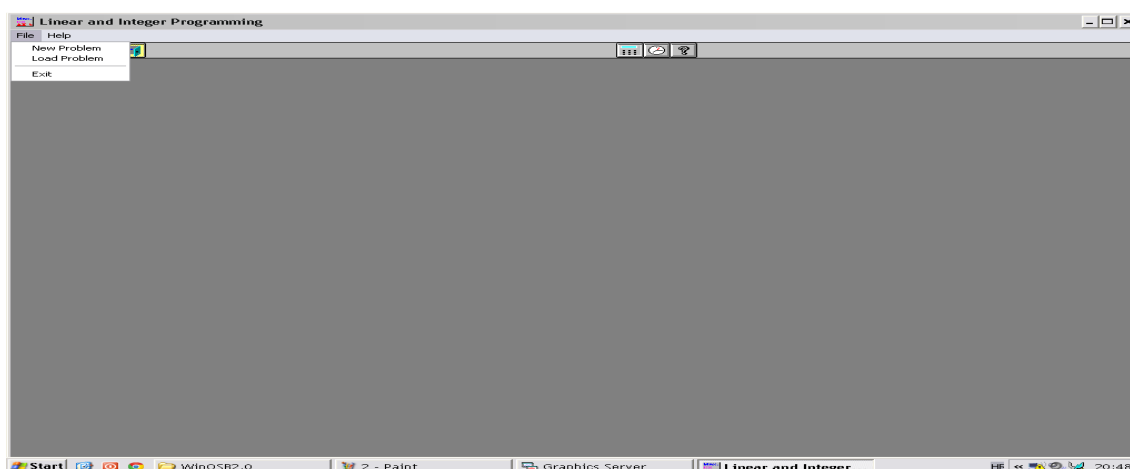
Da bi se moglo pristupiti rješavanju određenoga problema pomoću linearnoga i cjelobrojnoga linearnoga programiranja, nakon pokretanja računalnoga programa WinQSB potrebno je pokrenuti aplikacijski modul za linearno i cjelobrojno linearno programiranje klikom na aplikacijski model označen na Slici 3.1..

Slika 3.1. Pokretanje aplikacijskog modula za linearno i cjelobrojno linearno programiranje



Nakon pokretanja potprograma *Linear and Integer Programming* kliknemo na „File“ i odaberemo opciju „New Problem“ (kao što je prikazano na Slici 3.2.). Time otpočinjemo rješavanje problema.

Slika 3.2. Odabir opcije za rješavanje novog zadatka



U dobivenu tablicu (prikazanu na Slici 3.3.) upisujemo podatke. U pravokutnik pored natpisa *Problem Title* treba upisati naziv problema (npr. Problem 1, Primjer 1 i sl). U pravokutnik pored natpisa *Number of Variables* treba upisati ukupan broj nezavisnih varijabli, a u pravokutnik pored natpisa *Number of constraints* treba upisati ukupan broj uvjeta.

Ovisno o specifičnostima razmatranoga problema, bira se sljedeće opcije:

- ako se u problemu zahtijeva maksimizacija funkcije cilja, kliknemo na kružić pored natpisa *Maximization*;

- ako se u problemu zahtijeva minimizacija funkcije cilja, kliknemo na kružić pored natpisa *Minimization*;

- ako vrijednosti varijabli u matematičkom modelu mogu biti bilo koji (pozitivni ili negativni) decimalni brojevi, iracionalni brojevi i sl., odnosno ako se u modelu pojavljuje najmanje jedna varijabla čije vrijednosti mogu biti strogo negativne, kliknemo na kružić pored natpisa *Unsigned/unrestricted*;

- ako vrijednosti varijabli u matematičkom modelu mogu biti nenegativni decimalni brojevi, nenegativni iracionalni brojevi i sl., odnosno ako se u modelu pojavljuje najmanje jedna necjelobrojna varijabla, kliknemo na kružić pored natpisa *Nonnegative continuous*;

- ako su vrijednosti svih varijabli u matematičkom modelu mogu biti isključivo nenegativni cijeli brojevi, kliknemo na kružić pored natpisa *Nonnegative integer*;

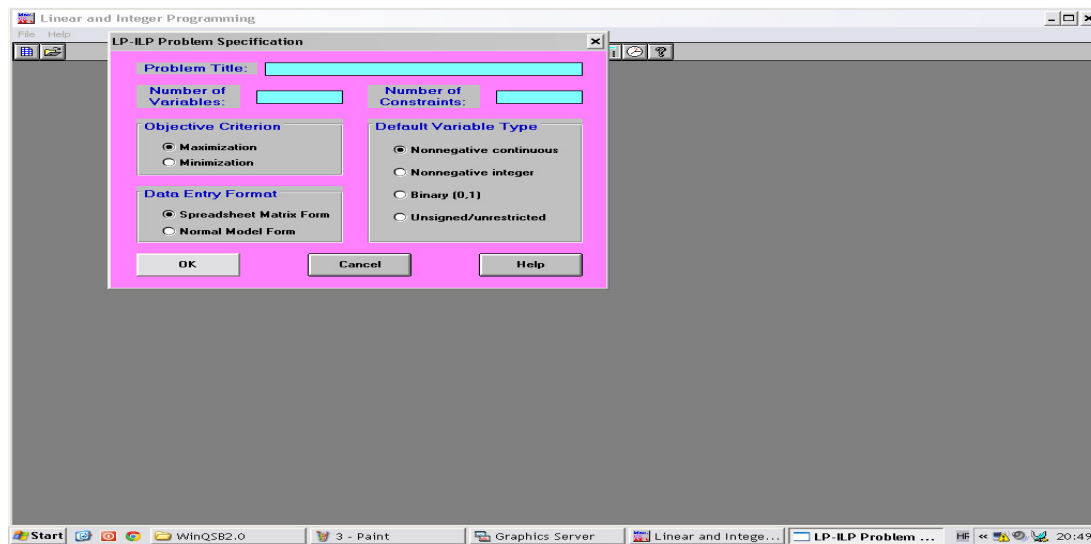
– ako vrijednosti svih varijabli u matematičkom modelu mogu biti isključivo elementi skupa $\{0, 1\}$, tj. ako se u matematičkom modelu pojavljuju isključivo tzv. varijable odlučivanja, kliknemo na kružić pored natpisa *Binary* (0, 1);

- želimo li polazne podatke unijeti u obliku proširene tablice zapisane u matricnom obliku (slično kao proračunske tablice MS Excela), kliknemo na kružić pored natpisa *Spreadsheet Matrix Form*;

– želimo li polazne podatke unijeti u obliku „obične“ tablice, odnosno upravo u onom obliku u kakvom su postavljeni u matematičkom modelu, kliknemo na kružić pored natpisa *Normal Model Form*.

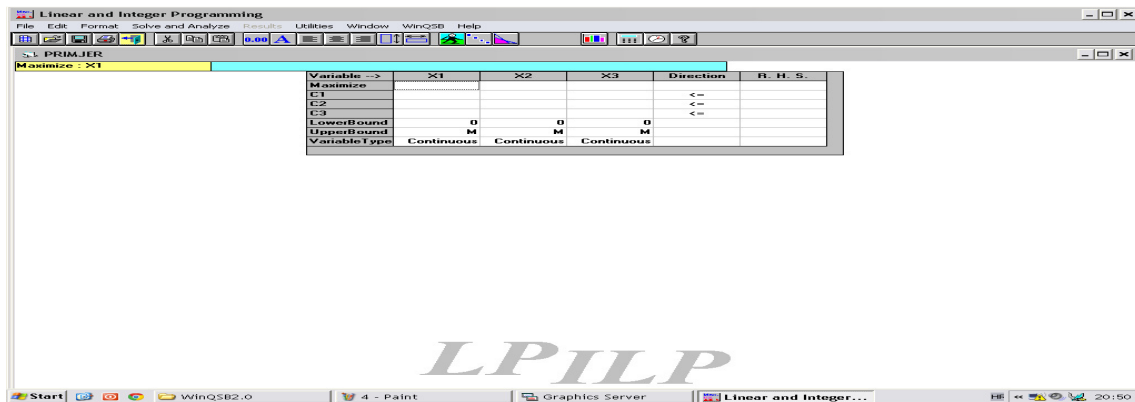
Unaprijed zadane postavke su *Maximization*, *Spreadsheet Matrix Form* i *Nonnegative continuous*. U praksi se, naime, najčešće javljaju problemi maksimizacije funkcije cilja čije su varijable nenegativni realni brojevi. Zapis funkcije cilja i uvjeta u matricnom obliku lakši je i pregledniji od zapisa u „običnom“ tabličnom obliku.

Slika 3.3. Tablica za unos podataka o matematičkom modelu



Nakon unosa podataka klikom na „OK“ dobivamo tablicu za unos koeficijenata prikazanu na Slici 3.4.

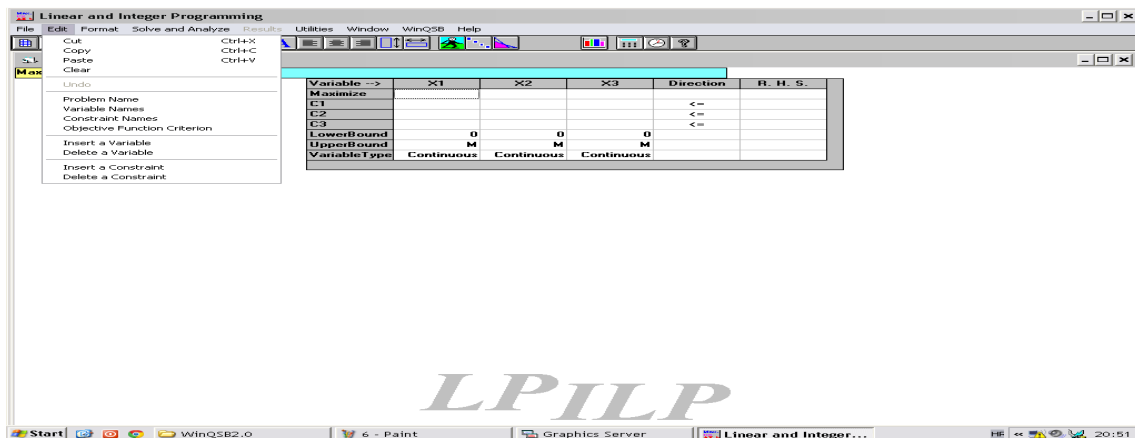
Slika 3.4. Tablica za upis koeficijenata



U retku *Variable* upisani su nazivi nezavisnih varijabli. WinQSB pretpostavlja da su varijable označene oznakama X1, X2 itd. U retku *Maximize/Minimize* treba upisati koeficijente funkcije cilja tako da prvi koeficijent bude onaj uz varijablu X1, drugi onaj uz varijablu X2 itd.

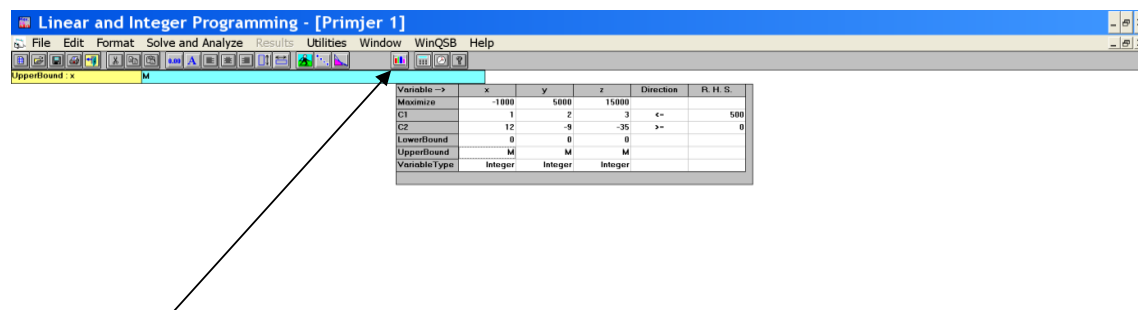
Klikom na opciju „*Edit*“ u glavnom izborniku otvara se padajući izbornik koji nudi opcije za preimenovanje varijable, dodavanje novih uvjeta, brisanje postojećih uvjeta, dodavanje novih varijabli i brisanje postojećih varijabli (kao što je prikazano na Slici 3.5.). Kroz primjere u nastavku rada objasniti će se korištenje spomenutih opcija.

Slika 3.5. Izbornik „*Edit*“ i njegove opcije



Opciju za rješavanje problema moguće je odabrati klikom na ikonicu (kao što je prikazano na Slici 3.6.) ili klikom na izbornik „*Solve and Analyze*“ i odabirom opcije „*Solve the Problem*“ (kao što je prikazano na Slici 3.7.).

Slika 3.6. Odabir opcije za rješavanje problema klikom na ikonicu



LPILP

Matrix Form You may double click to change a direction or variable type.

Slika 3.7. Odabir opcije „Solve and Analyze“ za rješavanje problema



Ako je pronađeno optimalno rješenje, pojavljuje se obavijest prikazana na Slici 3.8. Klikom miša na „OK“ dobiva se tablica s optimalnim rješenjima.

Slika 3.8. Obavijest da je rješenje nađeno



Klikom miša na „OK“ dobiva se tablica s optimalnim rješenjima prikazana na Slici 3.9.

Slika 3.9. Izlazna tablica

Linear and Integer Programming						
File Format Results Utilities Window Help						
Combined Report for						
	00:07:53		Friday	August	23	2013
	Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status
1	X1					at bound
2	X2					at bound
3	X3					basic
	Objective	Function	(Max.) =			
	Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price
1	C1		<=			
2	C2		>=			

U prvom stupcu izlazne tablice ispisuju se nazivi nezavisnih varijabli.

U drugom stupcu ispisuju se optimalne vrijednosti nezavisnih varijabli.

U retku *Objective Function* ispisuje se optimalna vrijednost funkcije cilja.

U stupcu *Constraints* ispisuju se oznake uvjeta redoslijedom kojim su upisani u tablicu ulaznih podataka.

U stupcu *Left Hand Side* ispisuju se vrijednosti koje se dobiju uvrštavanjem optimalnih vrijednosti nezavisnih varijabli u svaki pojedini uvjet.

U stupcu *Right Hand Side* ispisuje se vrijednost na desnoj strani svakoga pojedinoga uvjeta.

U stupcu *Slack or Surplus* ispisuju se razlike vrijednosti iz stupca *Left Hand Side* i vrijednosti iz stupca *Right Hand Side*. Ako je ta razlika jednaka 0, uvjet je zadovoljen tako da vrijedi znak jednakosti. Vrijednost različita od nule znači da postoji "suvišak" (ako uvjet sadrži znak \geq), odnosno "manjak" (ako uvjet sadrži znak \leq). Tada se znakovi \geq i \leq praktički mogu zamijeniti s $>$ i $<$.

Ovim je završen pregled osnovnih opcija potprograma *Linear and Integer Programming*. Korištenje tih opcija u rješavanju konkretnih primjera pokazat ćemo u sljedećem poglavlju.

4. PRIMJENA LINEARNOGA PROGRAMIRANJA NA PROBLEME PLANIRANJA PROIZVODNJE

4.1. Primjer 1

Pretpostavimo da neko postrojenje tvornice automobila „*Fordus* d.d.“ može napraviti jedan automobil marke *Ficho* u jednoj jedinici vremena (npr. jedan dan), jedan automobil marke *Speedy* u dvije jedinice vremena, a jedan automobil marke *Fasty* u tri jedinice vremena. Jedan automobil svake pojedine marke vozila troši prosječno redom 9.4 litre, 15.7 litara i 23.5 litara goriva na 100 km, a propisani prosjek potrošnje treba biti najviše 13 litara na 100 km. „*Fordus* d.d.“ ostvaruje gubitak od 1 000 n.j. na svakom proizvedenom *Fichi*, te dobit od 5 000 n.j. na svakom proizvedenom *Speedyju* i dobit od 15 000 n.j. na svakom proizvedenom *Fastyju*. Koliku maksimalnu dobit postrojenje može ostvariti proizvodnjom svih triju marki automobila u najviše 500 jedinica vremena?

Napomena: Nije nužno proizvesti barem jedan automobil svake pojedine marke.¹

Rješenje:

Pretpostavimo da treba proizvesti ukupno x automobila marke *Ficho*, y automobila marke *Speedy* i z automobila marke *Fasty*. Iznos dobiti koju postrojenje ostvaruje proizvodnjom y automobila marke *Speedy* jednak je $5\,000 \cdot y$ n.j. Iznos dobiti koju postrojenje ostvaruje proizvodnjom z automobila marke *Fasty* jednak je

¹ Primjer je napisan prema primjeru navedenom u [10].

15 000 · z n.j. Iznos gubitka kojega postrojenje ostvaruje proizvodnjom x automobila marke *Ficho* jednak je 1 000 · x n.j. Stoga je ukupan iznos dobiti nastale proizvodnjom svih triju marki automobila jednak:

$$-1\,000 \cdot x + 5\,000 \cdot y + 15\,000 \cdot z \text{ n.j.}$$

Za proizvodnju x automobila marke *Ficho* potrebno je ukupno 1 · x jedinica vremena. Za proizvodnju y automobila marke *Speedy* potrebno je ukupno 2 · y jedinica vremena. Za proizvodnju z automobila marke *Fasty* potrebno je ukupno 3 · z jedinica vremena. Stoga je za proizvodnju svih triju marki automobila potrebno ukupno $x + 2 \cdot y + 3 \cdot z$ jedinica vremena. To vrijeme ne smije biti strogo veće od 500 jedinica vremena, pa dobivamo uvjet :

$$x + 2 \cdot y + 3 \cdot z \leq 500$$

x automobila marke *Ficho* na 100 km potroše prosječno 9.4 · x litara goriva. y automobila marke *Speedy* na 100 km potroše prosječno 15.7 · y litara goriva. z automobila marke *Fasty* na 100 km potroše prosječno 23.5 · z litara goriva. Stoga svi proizvedeni automobili na 100 km potroše ukupno $9.4 \cdot x + 15.7 \cdot y + 23.5 \cdot z$ litara goriva. Budući da prosječna potrošnja goriva svih triju marki automobila mora biti najviše 13 litara na 100 km, dobivamo uvjet:

$$\frac{9.4 \cdot x + 15.7 \cdot y + 23.5 \cdot z}{x + y + z} \leq 13.$$

Taj uvjet možemo transformirati na sljedeći način:

$$\begin{aligned} 9.4 \cdot x + 15.7 \cdot y + 23.5 \cdot z &\leq 13 \cdot (x + y + z) \\ 9.4 \cdot x + 15.7 \cdot y + 23.5 \cdot z &\leq 13 \cdot x + 13 \cdot y + 13 \cdot z \\ 9.4 \cdot x + 15.7 \cdot y + 23.5 \cdot z - 13 \cdot x - 13 \cdot y - 13 \cdot z &\leq 0 \\ -3.6 \cdot x + 2.7 \cdot y + 10.5 \cdot z &\leq 0 \quad / \cdot \left(-\frac{10}{3}\right) \\ 12 \cdot x - 9 \cdot y - 35 \cdot z &\geq 0 \end{aligned}$$

Ovim uvjetima dodajemo prirodne uvjete na varijable x, y, z :

$$x, y, z \in \mathbf{N}_0 = \{0, 1, 2, 3, \dots\}$$

Nulu uključujemo zbog pretpostavke da nije nužno proizvesti barem jedan automobil svake pojedine marke. Logično je da ne možemo proizvesti npr. 1.23 automobila, pa zbog toga varijable x, y i z moraju biti nenegativni cijeli brojevi.

Stoga matematički model promatranoga problema glasi:

$$\text{maksimizirati } f(x, y, z) = -1\,000 \cdot x + 5\,000 \cdot y + 15\,000 \cdot z$$

pod uvjetima:

$$x + 2 \cdot y + 3 \cdot z \leq 500$$

$$12 \cdot x - 9 \cdot y - 35 \cdot z \geq 0$$

$$x, y, z \in \mathbf{N}_0$$

Pokrenimo potprogram *Linear and Integer Programming*. Odaberimo opciju „New problem“. U dobivenu tablicu upisujemo sljedeće podatke:

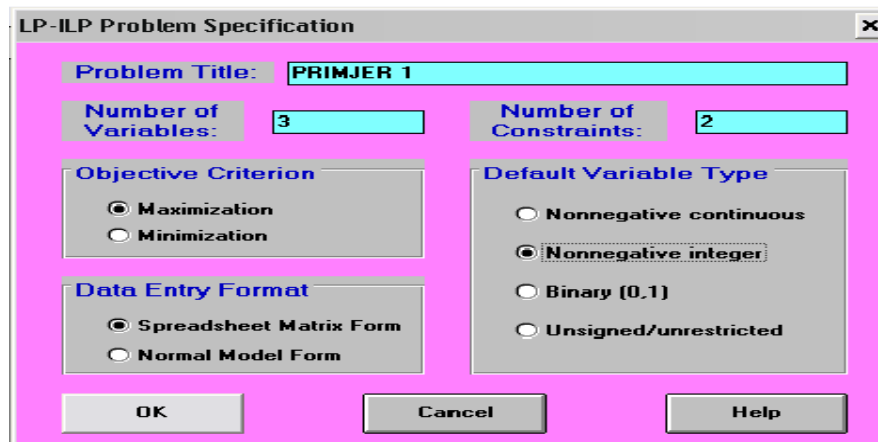
- naziv problema: Primjer 1;
- broj varijabli: 3 (u ovom primjeru su tri varijable);
- broj uvjeta: 2 (u ovom primjeru su dva uvjeta);

Potom kliknemo na kružice pored sljedećih natpisa:

- *Maximization* (u ovom primjeru se radi o maksimizaciji dobiti);
- *Nonnegative integer* (u ovom primjeru se radi o nenegativnim cijelim brojevima);
- *Spreadsheet Matrix Form* (ulazni podatci bit će prikazani u matričnom obliku).

Tako dobivamo tablicu prikazanu na Slici 4.1.

Slika 4.1. Unos podataka o matematičkom modelu za Primjer 1 (izradio autor)

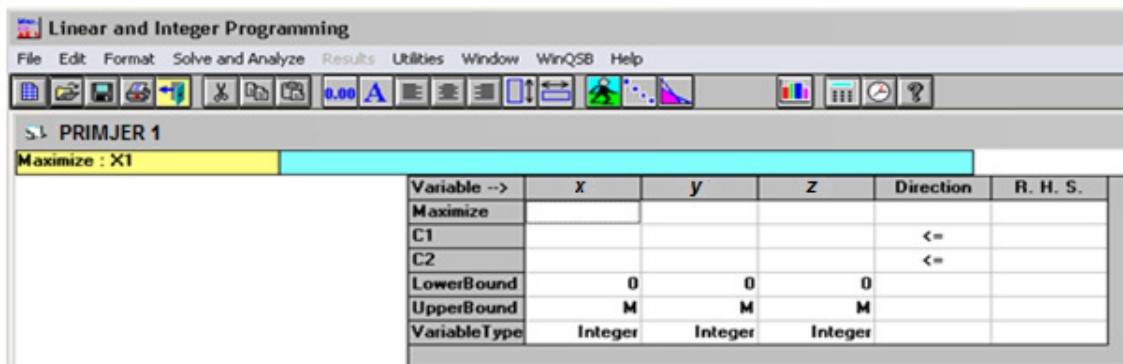


The dialog box 'LP-ILP Problem Specification' contains the following fields and options:

- Problem Title:** PRIMJER 1
- Number of Variables:** 3
- Number of Constraints:** 2
- Objective Criterion:**
 - ☒ Maximization
 - ☐ Minimization
- Data Entry Format:**
 - ☒ Spreadsheet Matrix Form
 - ☐ Normal Model Form
- Default Variable Type:**
 - ☐ Nonnegative continuous
 - ☒ Nonnegative integer
 - ☐ Binary {0,1}
 - ☐ Unsigned/unrestricted
- Buttons:** OK, Cancel, Help

Klikom na „OK“ otvara nam se tablica za upis koeficijenata uz varijable iz modela. Tablica je prikazana na Slici 4.2.

Slika 4.2. Tablica za upis koeficijenata iz Primjera 1 (izradio autor)



The window 'Linear and Integer Programming' displays the problem specification for 'PRIMJER 1'. The table below shows the coefficients for the objective function and constraints.

Variable -->	x	y	z	Direction	R. H. S.
Maximize					
C1				<=	
C2				<=	
LowerBound	0	0	0		
UpperBound	M	M	M		
VariableType	Integer	Integer	Integer		

Radi lakšeg upisivanja koeficijenata iz matematičkoga modela promijenjeni su nazivi varijabli u x, y i z korištenjem izbornika *Edit* i opcije *Variable Names*.

U tablicu je potrebno unijeti koeficijente uz varijable u funkciji cilja i uvjetima.

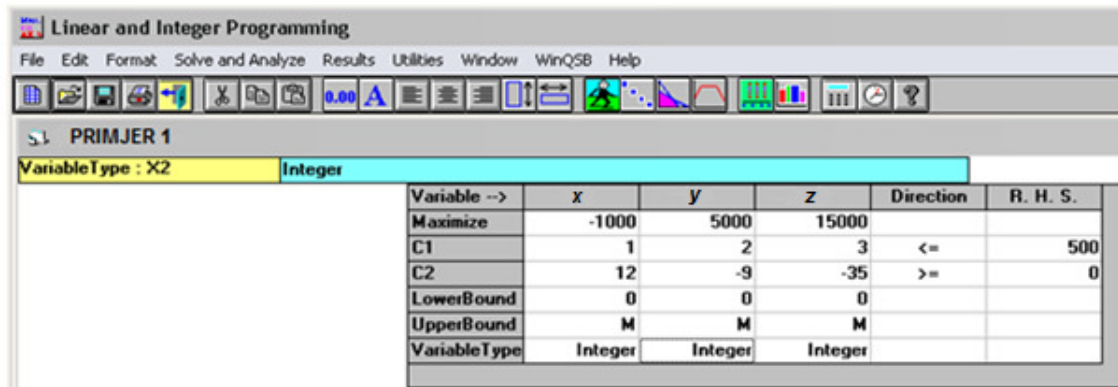
U redak *Maximize* unose se redom koeficijenti varijabli funkcije cilja: -1000, 5000, 15000.

U redak *C1* unose se koeficijenti uz varijable x, y i z u prvom uvjetu: 1, 2, 3, 500.

U redak *C2* unose se koeficijenti uz varijable x, y i z u drugom uvjetu: 12, -9, -35, 0. Pritom dvoklikom miša na znak <= promijenimo taj znak u >=. Korisno je pohraniti dobivenu tablicu kako bi se spriječio mogući gubitak podataka i omogućilo

ponovno korištenje ili eventualno mijenjanje podataka. Drugi korak prikazan je slikom 4.3.

Slika 4.3. Unos koeficijenata funkcije cilja i uvjeta u Primjeru 1 (izradio autor)



Radi rješavanja Primjera 2 pohranimo dobiveni model pod nazivom *Primjer 1*. Kliknemo na ikonicu diskete, pa u pravokutnik ispod natpisa *File name* (Naziv datoteke) upišemo *Primjer1.lpp*. Potom kliknemo na *OK*.

Klikom na ikonicu sa Slike 3.6. dobiva se tablica sa rješenjima prikazana na Slici 4.4. Iz nje treba očitati optimalno rješenje ovoga problema, te optimalnu vrijednost funkcije cilja.

Optimalne vrijednosti varijabli x , y i z očitavamo iz stupca *Solution Value*, a optimalnu vrijednost funkcije cilja iz retka *Objective Function (Max.)* =.

Slika 4.4. Tablica s optimalnim vrijednostima za Primjer 1. (izradio autor)

		00:07:53	Friday	August	23	2013
Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	
1	x	246,0000	-1,000,0000	-246,000,0000	-1,000,0000	at bound
2	y	1,0000	5,000,0000	5,000,0000	-5,000,0000	at bound
3	z	84,0000	15,000,0000	1,260,000,0000	0	basic
Objective Function		[Max.] = 1,019,000,0000				
Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	
1	C1	500,0000	<=	500,0000	0	5,000,0000
2	C2	3,0000	>=	0	3,0000	0

Optimalno rješenje ovoga problema je $(x^*, y^*, z^*) = (246, 1, 84)$, a optimalna vrijednost funkcije cilja iznosi 1 019 000. Dakle, treba proizvesti 246 automobila marke *Ficho*, 1 automobil marke *Speedy* i 84 automobila marke *Fasty*. Optimalna dobit iznosi 1 019 000 novčanih jedinica.

U sljedećem primjeru provedena je analiza osjetljivosti optimalnoga rješenja (utvrđivanje što se dogodi s optimalnim rješenjem ako se doda neki novi uvjet ili malo promijeni neku vrijednost u postojećim uvjetima).

4.2. Primjer 2

Riješimo Primjer 1., ali bez pretpostavke o cjelobrojnosti broja proizvedenih automobila. (Pretpostavka o nenegativnosti ostaje valjana.) Usporedimo dobivene optimalne vrijednosti funkcije cilja.

Rješenje:

Matematički je model gotovo isti kao i u Primjeru 1. Mijenja se samo posljednji od triju navedenih uvjeta. Vrijednosti varijabli x , y i z mogu biti i decimalni brojevi, pa novi matematički model glasi:

maksimizirati $f(x, y, z) = -1000 \cdot x + 5000 \cdot y + 15\,000 \cdot z$

pod uvjetima:

$$x + 2 \cdot y + 3 \cdot z \leq 500$$

$$12 \cdot x - 9 \cdot y - 35 \cdot z \geq 0$$

$$x, y, z \in \langle 0, +\infty \rangle$$

Prethodni zadatak koji je pohranjen u memoriji dovoljno je samo ponovno otvoriti koristeći opciju „Load problem“ koja se nalazi u izborniku „File“. Dvostrukim klikom na ćelije u retku „Variable Type“ mijenja se tip varijable. Budući da, prema pretpostavci primjera, vrijednost varijable može biti bilo koji strogo pozitivan realan broj, tip varijable mijenjamo iz *Integer* (cjelobrojna) u *Continuous* (neprekidna).

Slika 4.5. Tablica s koeficijentima matematičkoga modela za Primjer 2 (izradio autor)

The screenshot shows the 'Linear and Integer Programming' window. The title bar includes 'File', 'Edit', 'Format', 'Solve and Analyze', 'Results', 'Utilities', 'Window', 'WinQSB', and 'Help'. The menu bar contains icons for file operations, solving, and results. The main area is titled 'PRIMJER 1' and shows 'Variable Type : X3' and 'Continuous'. Below this is a table with the following data:

Variable -->	X	Y	Z	Direction	R. H. S.
Maximize	-1000	5000	15000		
C1	1	2	3	<=	500
C2	12	-9	-35	>=	0
LowerBound	0	0	0		
UpperBound	M	M	M		
VariableType	Continuous	Continuous	Continuous		

Na način opisan u prethodnom primjeru dobivamo novo rješenje prikazano na Slici 4.6.

Slika 4.6. Tablica s optimalnim vrijednostima za Primjer 2 (izradio: autor)

The screenshot shows the 'Combined Report for PRIMJER 1' window. The title bar includes 'File', 'Format', 'Results', 'Utilities', 'Window', and 'Help'. The menu bar contains icons for file operations, solving, and results. The main area displays the following report:

		00:14:43	Friday	August	23	2013		
Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	Allowable Min. c(j)	Allowable Max. c(j)	
1	X	246,4789	-1.000,0000	-246.478,9000	0	basic	-3.255,8140	5.000,0000
2	Y	0	5.000,0000	0	-1.366,1970	at bound	-M	6.366,1970
3	Z	84,5070	15.000,0000	1.267.606,0000	0	basic	12.060,6100	M
Objective	Function	[Max.] =	1.021.127,0000					
Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	Allowable Min. RHS	Allowable Max. RHS	
1	C1	500,0000	<=	500,0000	0	2.042,2540	0	M
2	C2	-0,0001	>=	0	0	-253,5211	-5.833,3340	6.000,0000

Optimalno rješenje ovoga problema je $(x^*, y^*, z^*) = (246.4789, 0, 84.5070)$, a optimalna vrijednost funkcije cilja iznosi 1 021 127. U ovom slučaju optimalna dobit iznosi 1 021 127 novčanih jedinica i za 2 127 novčanih jedinica, odnosno za $\frac{2\ 127}{1\ 019\ 000} \cdot 100 \approx 0.21\%$ je veća u odnosu na optimalnu dobit iz Primjera 1.

Važno je napomenuti da se zaokruživanjem optimalnih vrijednosti varijabli x , y i z na najbliži cijeli broj ne dobije optimalno rješenje iz Primjera 1., nego rješenje $(x^*, y^*, z^*) = (246, 0, 84)$. Zbog toga pretpostavka o cjelobrojnosti utječe na vrijednosti optimalnoga rješenja.

Iz gornje tablice moguće je očitati da su vrijednosti u stupcu *Slack or Surplus* jednake nuli. To znači da je u svakom uvjetu znak nejednakosti moguće zamijeniti znakom =, a da se dobiveno optimalno rješenje pritom ne izmijeni. U Primjeru 1. takvu zamjenu nije moguće napraviti jer se u drugom uvjetu pojavljuje „suvišak“ jednak 3.

4.3. Primjer 3

Tvrtka “*Kosilice* d.o.o.” bavi se proizvodnjom kosilica čija završna faza izradbe zahtijeva montažu triju dijelova. Svaki od tih triju dijelova može biti proizveden u bilo kojemu od dva odjela tvrtke. Osnovni podatci navedeni su u sljedećoj tablici.

Tablica 4.1. Osnovni podatci za Primjer 3.

Proizvodna stopa (jedinica/sat)				
	Dio 1	Dio 2	Dio 3	Trošak (n.j./sat)
Odjel 1	7	6	9	25.0
Odjel 2	6	11	5	12.5

U jednom radnom tjednu potrebno je proizvesti barem 1 050 proizvoda, pri čemu je moguće proizvesti najviše 1 200 proizvoda. Odjel 1 ima na raspolaganju 100 radnih sati a Odjel 2 110 radnih sati. Zbog organizacije proizvodnje nije dozvoljeno da jedan odjel proizvodi točno jedan od triju dijelova.

Potrebno je minimizirati tjedne troškove proizvodnje, pri čemu broj odrađenih radnih sati u svakom odjelu mora biti nenegativan cijeli broj. Pretpostavlja se da svaki odjel ne mora odraditi niti jedan radni sat, kao i da je skladište tvrtke dovoljno veliko da se u njemu može pohraniti cjelokupna tjedna proizvodnja svih triju dijelova.²

² Primjer je napisan prema primjeru iz [5].

Rješenje

Pri formiranju matematičkoga modela treba razmotriti razdiobu proizvodnje dijelova prema odjelima, ali i ukupan broj proizvedenih proizvoda na kraju tjedna. Stoga označimo:

x_1 – ukupan broj radnih sati odrađenih u Odjelu 1;

x_2 – ukupan broj radnih sati odrađenih u Odjelu 2;

x_3 – ukupan broj završenih proizvoda.

Odjel 1 ima na raspolaganju 100 radnih sati, a Odjel 2 110 radnih sati. Broj odrađenih radnih sati u svakom odjelu treba biti nenegativan cijeli broj. Stoga moraju vrijediti nejednakosti:

$$0 \leq x_1 \leq 100,$$

$$0 \leq x_2 \leq 110,$$

$$x_1, x_2 \in \mathbf{N}_0$$

Nadalje, prema uvjetima zadatka, treba proizvesti najmanje 1 050 proizvoda, ali nije moguće proizvesti strogo više od 1 200 proizvoda. I ukupan broj proizvoda treba biti prirodan broj, pa dobivamo uvjete:

$$1\,050 \leq x_3 \leq 1\,200,$$

$$x_3 \in \mathbf{N}$$

U Odjelu 1 proizvede se ukupno $7 \cdot x_1$ komada dijela 1, $6 \cdot x_1$ komada dijela 2 i $9 \cdot x_1$ komada dijela 3. Analogno, u Odjelu 2 proizvede se ukupno $6 \cdot x_2$ komada dijela 1, $11 \cdot x_2$ komada dijela 2 i $5 \cdot x_2$ komada dijela 3. Da bismo dobili barem x_3 završena komada proizvoda, ukupan broj proizvedenih komada svakoga pojedinoga dijela treba biti barem jednak x_3 , tj. trebamo barem x_3 komada dijela 1, barem x_3 komada dijela 2 i barem x_3 komada dijela 3. Tako dobivamo uvjete:

$$7 \cdot x_1 + 6 \cdot x_2 \geq x_3,$$

$$6 \cdot x_1 + 11 \cdot x_2 \geq x_3,$$

$$9 \cdot x_1 + 5 \cdot x_2 \geq x_3,$$

koje možemo zapisati u obliku

$$\begin{aligned}7 \cdot x_1 + 6 \cdot x_2 - x_3 &\geq 0, \\6 \cdot x_1 + 11 \cdot x_2 - x_3 &\geq 0, \\9 \cdot x_1 + 5 \cdot x_2 - x_3 &\geq 0.\end{aligned}$$

Ukupan broj svih proizvedenih dijelova jednak je:

$$(7 \cdot x_1 + 6 \cdot x_2) + (6 \cdot x_1 + 11 \cdot x_2) + (9 \cdot x_1 + 5 \cdot x_2) = 22 \cdot x_1 + 22 \cdot x_2.$$

Napokon, ukupan tjedni trošak rada Odjela 1 iznosi $25 \cdot x_1$ n.j., a ukupan tjedni trošak rada Odjela 2 iznosi $12.5 \cdot x_2$ n.j. Stoga je ukupan tjedni trošak rada obaju odjela jednak $25 \cdot x_1 + 12.5 \cdot x_2$ n.j. Taj trošak treba minimizirati.

Dakle, matematički model promatranoga problema glasi:

$$\text{minimizirati } f(x_1, x_2) = 25 \cdot x_1 + 12.5 \cdot x_2$$

pod uvjetima

$$\begin{aligned}7 \cdot x_1 + 6 \cdot x_2 - x_3 &\geq 0, \\6 \cdot x_1 + 11 \cdot x_2 - x_3 &\geq 0, \\9 \cdot x_1 + 5 \cdot x_2 - x_3 &\geq 0, \\0 \leq x_1 &\leq 100, \\0 \leq x_2 &\leq 110, \\1\,050 \leq x_3 &\leq 1\,200, \\x_1, x_2 &\in \mathbf{N}_0, \\x_3 &\in \mathbf{N}.\end{aligned}$$

Kliknemo na padajući izbornik „File“ i na njemu odaberemo opciju „New problem“. U dobivenu tablicu upisujemo sljedeće podatke:

- naziv problema: Primjer 3;
- broj varijabli: 3 (u ovom primjeru su tri varijable);
- broj uvjeta: 3 (u ovom primjeru su tri uvjeta);

Potom kliknemo na kružice pored sljedećih natpisa:

- *Minimization* (u ovom primjeru se radi o minimizaciji troškova);
- *Nonnegative integer* (u ovom primjeru se radi o nenegativnim cijelim brojevima);
- *Spreadsheet Matrix Form* (ulazni podatci bit će prikazani u matričnom obliku).

Tako dobivamo tablicu prikazanu na Slici 4.7.

Slika 4.7. Unos podataka o matematičkom modelu za Primjer 3. (izradio autor)

LP-ILP Problem Specification

Problem Title: PRIMJER 3

Number of Variables: 3 Number of Constraints: 3

Objective Criterion

☐ Maximization
☒ Minimization

Data Entry Format

☒ Spreadsheet Matrix Form
☐ Normal Model Form

Default Variable Type

☐ Nonnegative continuous
☒ Nonnegative integer
☐ Binary (0,1)
☐ Unsigned/unrestricted

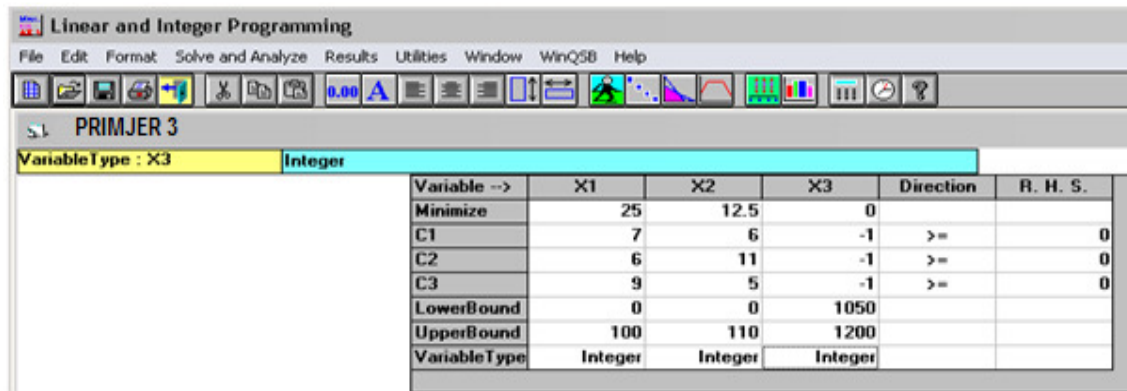
OK Cancel Help

Kliknemo na OK. U dobivenu tablicu je potrebno unijeti koeficijente uz varijable x_1 , x_2 i x_3 u funkciji cilja i uvjetima. Upisujemo:

- u redak *Minimize*: 25, 12.5, 0
- u redak *C1*: 7, 6, -1
- u redak *C2*: 6, 11, -1
- u redak *C3*: 9, 5, -1
- u redak *Lower Bound*: 0, 0, 1050 (najmanja vrijednost varijable x_3 je 1050)
- u redak *Upper Bound*: 100, 110, 1200

Dvostrukim klikom miša na svaki od triju znakova \leq promijenimo te znakove u \geq . Tako dobivamo tablicu prikazanu na Slici 4.8. Potrebno je pohraniti dobivenu tablicu kako bi se omogućilo ponovno korištenje ili eventualno mijenjanje podataka. Pohranimo je pod nazivom *Primjer3.lpp*.

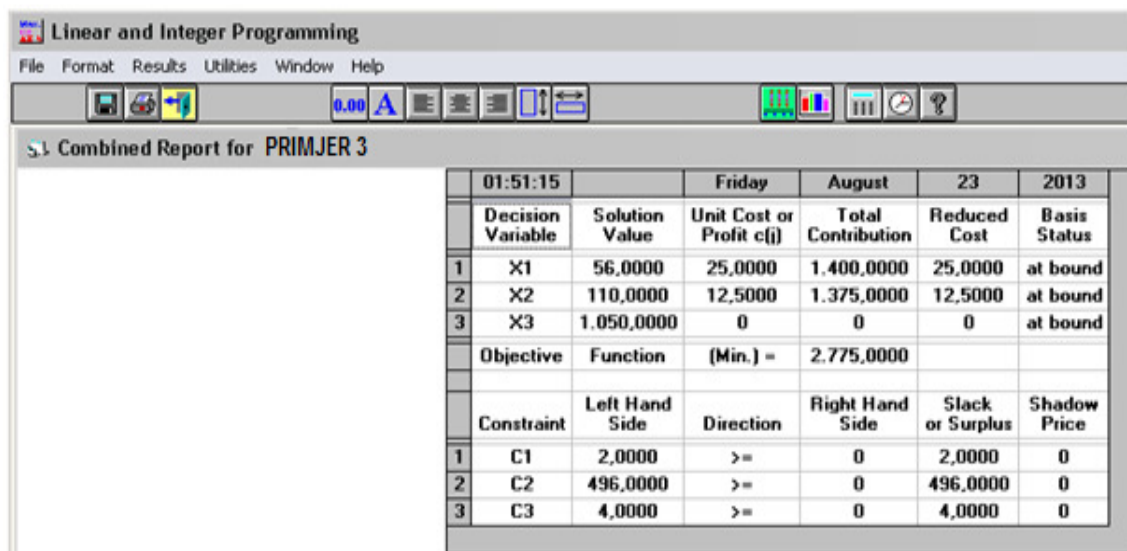
Slika 4.8. Tablica s koeficijentima matematičkoga modela za Primjer 3. (izradio autor)



Variable -->	X1	X2	X3	Direction	R. H. S.
Minimize	25	12.5	0		
C1	7	6	-1	>=	0
C2	6	11	-1	>=	0
C3	9	5	-1	>=	0
LowerBound	0	0	1050		
UpperBound	100	110	1200		
VariableType	Integer	Integer	Integer		

Ranije opisanim postupkom dobivamo tablicu s optimalnim vrijednostima prikazanu na Slici 4.9.

Slika 4.9. Tablica s optimalnim vrijednostima za Primjer 3 (izradio autor)



01:51:15		Friday	August	23	2013
Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status
1 X1	56,0000	25,0000	1,400,0000	25,0000	at bound
2 X2	110,0000	12,5000	1,375,0000	12,5000	at bound
3 X3	1,050,0000	0	0	0	at bound
Objective	Function	(Min.) =	2,775,0000		
Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price
1 C1	2,0000	>=	0	2,0000	0
2 C2	496,0000	>=	0	496,0000	0
3 C3	4,0000	>=	0	4,0000	0

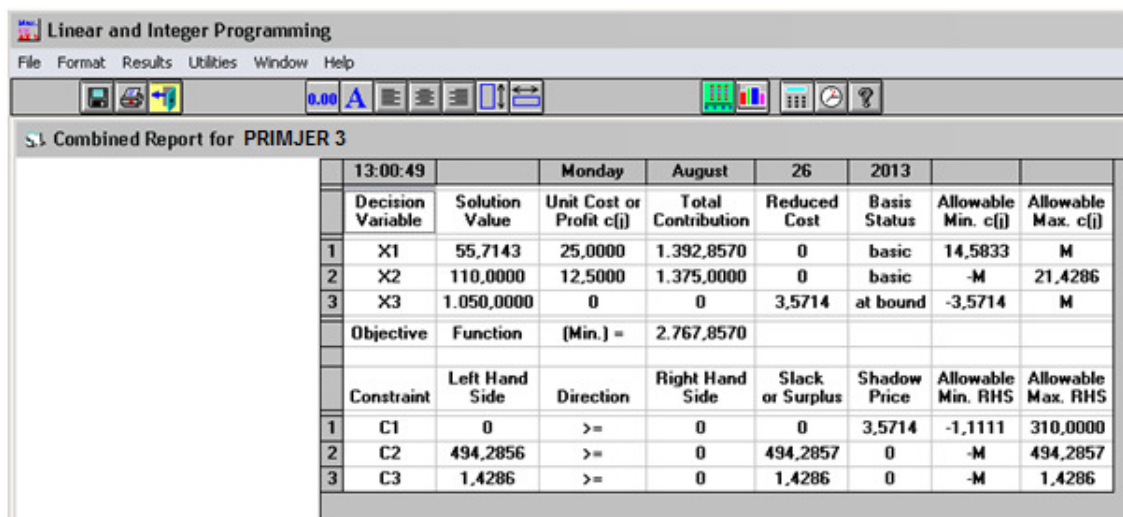
Iz tablice vidimo da je optimalno rješenje ovoga problema $(x_1^*, x_2^*, x_3^*) = (56, 110, 1\ 050)$, dok optimalna vrijednost funkcije cilja iznosi 2 775.

Dakle, Odjel 1 tjedno treba raditi 56 radnih sati, a Odjel 2 svih 110 radnih sati. Optimalan tjedni ukupan broj proizvedenih proizvoda je 1 050. Minimalni ukupni troškovi iznose 2 775 n.j.

Kada bi se, analogno kao u Primjeru 2., zanemarila pretpostavka o cjelobrojnosti varijabli, odnosno promijenio tip varijable iz *Integer* u *Continuous*, dobili bismo optimalno rješenje $(x_1^*, x_2^*, x_3^*) = (55.7143, 110, 1.050)$ i optimalnu vrijednost funkcije cilja 2 767.8570 n.j. (vidjeti Sliku 4.10.)

U ovom se slučaju zaokruživanjem optimalnih vrijednosti varijabli (ali ne i optimalne vrijednosti funkcije cilja) na najbliži prirodan broj dobiva optimalno rješenje iz Primjera 3.

Slika 4.10. Tablica s optimalnim vrijednostima za Primjer 3. uz promjenu tipa varijabli (izradio autor)



13:00:49 Monday August 26 2013								
	Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	Allowable Min. c(j)	Allowable Max. c(j)
1	X1	55,7143	25,0000	1.392,8570	0	basic	14,5833	M
2	X2	110,0000	12,5000	1.375,0000	0	basic	-M	21,4286
3	X3	1.050,0000	0	0	3,5714	at bound	-3,5714	M
	Objective Function		[Min.] =	2.767,8570				
	Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	Allowable Min. RHS	Allowable Max. RHS
1	C1	0	>=	0	0	3,5714	-1,1111	310,0000
2	C2	494,2856	>=	0	494,2857	0	-M	494,2857
3	C3	1,4286	>=	0	1,4286	0	-M	1,4286

Uz pretpostavku o cjelobrojnosti varijabli, za sva tri uvjeta dobivamo „suvišak“ (što znači da znak nejednakosti ne smijemo zamijeniti znakom jednakosti), dok uz pretpostavku da su varijable nenegativni realni brojevi za prvi uvjet dobivamo „suvišak“ jednak 0. Stoga se može zaključiti da u tom slučaju u prvom uvjetu znak \geq možemo zamijeniti znakom $=$, a da se vrijednosti optimalnih rješenja ne izmijene. Za ostala dva uvjeta ta tvrdnja očito ne vrijedi, ali treba primijetiti da u drugom slučaju za te uvjete dobivamo manje „suviškove“ nego u prvom slučaju.

4.4. Primjer 4

Riješimo prethodni primjer uz dodatan uvjet da odlukom uprave tvrtke oba odjela moraju tjedno raditi jednak broj radnih sati. Odredimo relativnu promjenu optimalne vrijednosti svake pojedine varijable (ako je to moguće), kao i relativnu promjenu optimalne vrijednosti funkcije cilja.

Rješenje:

Matematički je model gotovo isti kao i u Primjeru 3. Naznačenim uvjetima dodajemo još jedan. Oba odjela trebaju odraditi jednak tjedni broj radnih sati. To znači da mora vrijediti jednakost:

$$x_1 = x_2,$$

odnosno ekvivalentna jednakost

$$x_1 - x_2 = 0.$$

Stoga novi matematički model promatranoga problema glasi:

$$\text{minimizirati } f(x_1, x_2) = 25 \cdot x_1 + 12.5 \cdot x_2$$

pod uvjetima

$$7 \cdot x_1 + 6 \cdot x_2 - x_3 \geq 0,$$

$$6 \cdot x_1 + 11 \cdot x_2 - x_3 \geq 0,$$

$$9 \cdot x_1 + 5 \cdot x_2 - x_3 \geq 0,$$

$$x_1 - x_2 = 0,$$

$$0 \leq x_1 \leq 100,$$

$$0 \leq x_2 \leq 110,$$

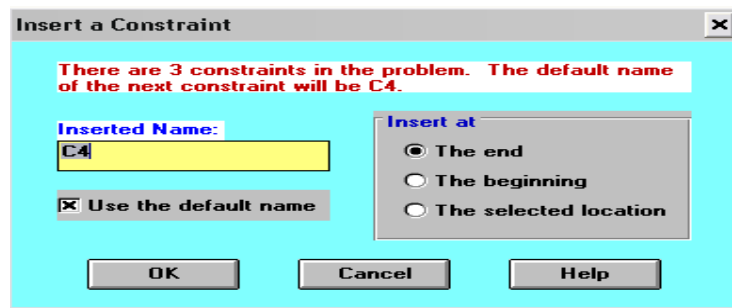
$$1\,050 \leq x_3 \leq 1\,200,$$

$$x_1, x_2 \in \mathbf{N}_0,$$

$$x_3 \in \mathbf{N}.$$

Tablicu s koeficijentima matematičkoga modela u prethodnom primjeru otvorimo preko opcije „Load problem“. Kliknemo na izbornik „Edit“ i odaberemo opciju *Insert a Constraint* koja omogućuje dodavanje novoga uvjeta. Otvara se izbornik prikazan na slici 4.11. Novi uvjet označimo s C4 i dodajemo ga na kraj tablice.

Slika 4.11. Izbornik za dodavanje novoga uvjeta (izradio autor)



Kliknemo na OK. U redak C4 upisujemo koeficijente uz varijable x_1 i x_2 . To su redom 1 i -1. U istom retku znak \leq mijenjamo u $=$ tako da u stupcu *Direction* upišemo $=$. Svi ostali podatci ostaju neizmijenjeni. Dobivamo tablicu prikazanu na Slici 4.12.

Slika 4.12. Tablica s koeficijentima matematičkoga modela za Primjer 4 (izradio autor)

Linear and Integer Programming

File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

PRIMJER 3

VariableType : R.H.S.

Variable -->	X1	X2	X3	Direction	R. H. S.
Minimize	25	12.5	0		
C1	7	6	-1	\geq	0
C2	6	11	-1	\geq	0
C3	9	5	-1	\geq	0
C4	1	-1		$=$	0
LowerBound	0	0	1050		
UpperBound	100	110	1200		
VariableType	Integer	Integer	Integer		

Ranije opisanim postupkom dobivamo tablicu s optimalnim vrijednostima prikazanu na Slici 4.13.

Slika 4.13. Tablica s optimalnim vrijednostima za Primjer 4 (izradio autor)

Linear and Integer Programming						
File Format Results Utilities Window Help						
0.00						
Combined Report for PRIMJER 3						
	02:00:06		Friday	August	23	2013
Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	
1 X1	81,0000	25,0000	2.025,0000	37,5000	at bound	
2 X2	81,0000	12,5000	1.012,5000	0	basic	
3 X3	1.053,0000	0	0	0	basic	
Objective	Function	(Min.) =	3.037,5000			
Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	
1 C1	0	>=	0	0	0	
2 C2	324,0000	>=	0	324,0000	0	
3 C3	81,0000	>=	0	81,0000	0	
4 C4	0	=	0	0	-12,5000	

Optimalno rješenje ovoga problema je $(x_1^*, x_2^*, x_3^*) = (81, 81, 1\,053)$, a optimalna vrijednost funkcije cilja iznosi 3 037.5. Dakle, oba odjela tjedno trebaju raditi po 81 radni sat. Optimalan tjedni ukupan broj proizvedenih proizvoda je 1 053. Minimalni ukupni troškovi iznose 3 037.5 n.j.

Primijetimo da se optimalna vrijednost varijable x_1 povećala za $81 - 56 = 25$, odnosno za $\frac{25}{56} \cdot 100 \approx 44.6\%$. Optimalna vrijednost varijable x_2 smanjila se za $112 - 81 = 31$, odnosno za $\frac{31}{81} \cdot 100 \approx 38.27\%$. Optimalna vrijednost varijable x_3 povećala se za $1\,053 - 1\,050 = 3$, odnosno za $\frac{3}{1\,050} \cdot 100 \approx 0.3\%$. Optimalna vrijednost funkcije cilja povećala se za $3\,037.5 - 2\,775 = 262.5$, odnosno za $\frac{262.5}{2\,775} \cdot 100 \approx 9.5\%$.

4.5. Primjer 5

Tvrtka „Klimus d.o.o.“ proizvodi četiri tipa klima-uređaja. Završni dio proizvodnoga procesa sastoji se od operacija montaže, poliranja i pakiranja. U donjoj je tablici za svaki pojedini tip navedeno vrijeme potrebno za svaku pojedinu operaciju, kao i jedinična dobit.

Tablica 4.2. Osnovni podatci za Primjer 5.

TIP	MONTAŽA [minuta/komad]	POLIRANJE [minuta/komad]	PAKIRANJE [minuta/komad]	DOBIT [n.j./komad]
1	2	3	2	1.50
2	4	2	3	2.50
3	3	3	2	3.00
4	7	4	5	4.50

S obzirom na trenutno stanje radne snage, vlasnik tvrtke procjenjuje da godišnje na raspolaganju ima 100 000 minuta za operaciju montaže, 50 000 minuta za operaciju poliranja i 60 000 minuta za operaciju pakiranja. Treba napraviti godišnji plan proizvodnje tako da pripadna ukupna dobit bude maksimalna. (Moguće je ne proizvesti niti jedan komad određenoga tipa klima-uređaja.)³

Rješenje:

Označimo:

x_1 = broj proizvedenih komada klima-uređaja tipa 1;

x_2 = broj proizvedenih komada klima-uređaja tipa 2;

x_3 = broj proizvedenih komada klima-uređaja tipa 3;

x_4 = broj proizvedenih komada klima-uređaja tipa 4.

³Primjer je napisan prema primjeru iz [4]

Za montažu x_1 proizvedenih komada klima-uređaja tipa 1 potrebno je $2 \cdot x_1$ minuta.
 Za montažu x_2 proizvedenih komada klima-uređaja tipa 2 potrebno je $4 \cdot x_2$ minuta.
 Za montažu x_3 proizvedenih komada klima-uređaja tipa 3 potrebno je $3 \cdot x_3$ minuta.
 Za montažu x_4 proizvedenih komada klima-uređaja tipa 4 potrebno je $7 \cdot x_4$ minuta.
 Stoga je ukupno vrijeme potrebno za montažu svih četiriju tipova klima-uređaja $2 \cdot x_1 + 4 \cdot x_2 + 3 \cdot x_3 + 7 \cdot x_4$ minuta. Budući da za montažu imamo na raspolaganju najviše 100 000 minuta, dobivamo uvjet:

$$2 \cdot x_1 + 4 \cdot x_2 + 3 \cdot x_3 + 7 \cdot x_4 \leq 100\,000.$$

Za poliranje x_1 proizvedenih komada klima-uređaja tipa 1 potrebno je $3 \cdot x_1$ minuta.
 Za poliranje x_2 proizvedenih komada klima-uređaja tipa 2 potrebno je $2 \cdot x_2$ minuta.
 Za poliranje x_3 proizvedenih komada klima-uređaja tipa 3 potrebno je $3 \cdot x_3$ minuta.
 Za poliranje x_4 proizvedenih komada klima-uređaja tipa 4 potrebno je $4 \cdot x_4$ minuta.
 Stoga je ukupno vrijeme potrebno za poliranje svih četiriju tipova klima-uređaja $3 \cdot x_1 + 2 \cdot x_2 + 3 \cdot x_3 + 4 \cdot x_4$ minuta. Budući da za poliranje imamo na raspolaganju najviše 50 000 minuta, dobivamo uvjet:

$$3 \cdot x_1 + 2 \cdot x_2 + 3 \cdot x_3 + 4 \cdot x_4 \leq 50\,000.$$

Za pakiranje x_1 proizvedenih komada klima-uređaja tipa 1 potrebno je $2 \cdot x_1$ minuta.
 Za pakiranje x_2 proizvedenih komada klima-uređaja tipa 2 potrebno je $3 \cdot x_2$ minuta.
 Za pakiranje x_3 proizvedenih komada klima-uređaja tipa 3 potrebno je $2 \cdot x_3$ minuta.
 Za pakiranje x_4 proizvedenih komada klima-uređaja tipa 4 potrebno je $5 \cdot x_4$ minuta.
 Stoga je ukupno vrijeme potrebno za pakiranje svih četiriju tipova klima-uređaja $2 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + 5 \cdot x_4$ minuta. Budući da za pakiranje imamo na raspolaganju najviše 60 000 minuta, dobivamo uvjet:

$$2 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + 5 \cdot x_4 \leq 60\,000.$$

Dobit nastala prodajom x_1 proizvedenih komada klima-uređaja tipa 1 iznosi $1.5 \cdot x_1$ n.j. Dobit nastala prodajom x_2 proizvedenih komada klima-uređaja tipa 2 iznosi

$2.5 \cdot x_2$ n.j. Dobit nastala prodajom x_3 proizvedenih komada klima-uređaja tipa 3 iznosi $3 \cdot x_3$ n.j. Dobit nastala prodajom x_4 proizvedenih komada klima-uređaja tipa 4 iznosi $4.5 \cdot x_4$ n.j. Stoga ukupna dobit nastala prodajom svih četiriju tipova klima-uređaja iznosi $1.5 \cdot x_1 + 2.5 \cdot x_2 + 3 \cdot x_3 + 4.5 \cdot x_4$ n.j. Tu dobit treba maksimizirati.

Vrijednosti varijabli x_1 , x_2 , x_3 i x_4 su nenegativni cijeli brojevi jer dozvoljavamo da je moguće ne proizvesti niti jedan komad pojedinoga tipa klima-uređaja. Dakle,

$$x_1, x_2, x_3, x_4 \in \mathbf{N}_0$$

Tako smo dobili matematički model promatranoga problema:

$$\text{maksimizirati } f(x_1, x_2, x_3, x_4) = 1.5 \cdot x_1 + 2.5 \cdot x_2 + 3 \cdot x_3 + 4.5 \cdot x_4$$

pod uvjetima

$$2 \cdot x_1 + 4 \cdot x_2 + 3 \cdot x_3 + 7 \cdot x_4 \leq 100\,000,$$

$$3 \cdot x_1 + 2 \cdot x_2 + 3 \cdot x_3 + 4 \cdot x_4 \leq 50\,000,$$

$$2 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + 5 \cdot x_4 \leq 60\,000,$$

$$x_1, x_2, x_3, x_4 \in \mathbf{N}_0$$

Riješimo ovaj model koristeći WinQSB. Na ranije opisan način otvorimo novu datoteku, pa u njezine polazne podatke upišimo:

- naziv problema: Primjer 5;
- broj varijabli: 4 (u ovom primjeru su četiri varijable);
- broj uvjeta: 3 (u ovom primjeru su tri uvjeta);

Potom kliknemo na kružice pored sljedećih natpisa:

- *Maximization* (u ovom primjeru se radi o maksimizaciji ukupne dobiti);
- *Nonnegative integer* (u ovom primjeru se radi o nenegativnim cijelim brojevima);
- *Spreadsheet Matrix Form* (izlazni podatci bit će ispisani u matričnom obliku).

Tako dobivamo tablicu prikazanu na Slici 4.14.

Slika 4.14. Unos podataka o matematičkom modelu za Primjer 5. (izradio autor)

Kliknemo na OK. U dobivenu tablicu je potrebno unijeti koeficijente uz varijable x_1 , x_2 i x_3 u funkciji cilja i uvjetima. Upisujemo:

- u redak *Maximize*: 1.5, 2.5, 3, 4.5
- u redak *C1*: 2, 4, 3, 7, 100000
- u redak *C2*: 3, 2, 3, 4, 50000
- u redak *C3*: 2, 3, 2, 5, 60000

Tako dobivamo tablicu prikazanu na Slici 4.15. Potrebno je pohraniti dobivenu tablicu kako bi se omogućilo ponovno korištenje ili eventualno mijenjanje podataka. Pohranimo je pod nazivom *Primjer5.lpp*.

Slika 4.15. Tablica s koeficijentima matematičkoga modela za Primjer 5 (izradio autor)

Variable -->	X1	X2	X3	X4	Direction	R. H. S.
Maximize	1.5	2.5	3	4.5		
C1	2	4	3	7	<=	100000
C2	3	2	3	4	<=	50000
C3	2	3	2	5	<=	60000
LowerBound	0	0	0	0		
UpperBound	M	M	M	M		
VariableType	Integer	Integer	Integer	Integer		

Tablica s optimalnim vrijednostima dobivena uobičajenim postupkom prikazana je na Slici 4.16.

Slika 4.16. Tablica s optimalnim vrijednostima za Primjer 5 (izradio autor)

Linear and Integer Programming								
File Format Results Utilities Window Help								
S1 Combined Report for PRIMJER 5								
	02:21:18		Friday	August	23	2013		
	Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	Allowable Min. c(j)	Allowable Max. c(j)
1	X1	0	1,5000	0	-1,5000	at bound	-M	3,0000
2	X2	16.000,0000	2,5000	40.000,0000	0	basic	2,3571	4,5000
3	X3	6.000,0000	3,0000	18.000,0000	0	basic	2,5000	3,7500
4	X4	0	4,5000	0	-0,2000	at bound	-M	4,7000
	Objective	Function	(Max.) =	58.000,0000				
	Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	Allowable Min. RHS	Allowable Max. RHS
1	C1	82.000,0000	<=	100.000,0000	18.000,0000	0	82.000,0000	M
2	C2	50.000,0000	<=	50.000,0000	0	0,8000	40.000,0000	90.000,0000
3	C3	60.000,0000	<=	60.000,0000	0	0,3000	33.333,3400	75.000,0000

Iz dobivene tablice vidimo da je optimalno rješenje ovoga problema $(x_1^*, x_2^*, x_3^*, x_4^*) = (0, 16\ 000, 6\ 000, 0)$, dok optimalna vrijednost funkcije cilja iznosi 58 000. Dakle, treba proizvesti ukupno 16 000 komada klima-uređaja tipa 2 i 6 000 komada klima-uređaja tipa 3, a ne treba proizvesti niti jedan uređaj tipa 1, odnosno tipa 4. Optimalna ukupna dobit iznosi 58 000 n.j.

Potpuno analogno kao u prethodnim primjerima može se pokazati da se u ovome primjeru dobije potpuno isto optimalno rješenje i ako se pretpostavi da vrijednosti varijabli x_1 , x_2 , x_3 i x_4 nisu cjelobrojne, već samo nenegativne. Dakle, pretpostavka o cjelobrojnosti varijabli nema utjecaja na optimalno rješenje.

4.6. Primjer 6

Riješimo prethodni primjer uz dodatan uvjet da, zbog potreba izvoza, godišnje treba proizvesti najmanje 1 000 komada svakoga pojedinoga tipa klima-uređaja. Odredimo relativnu promjenu optimalne vrijednosti svake pojedine varijable (ako je to moguće), kao i relativnu promjenu optimalne vrijednosti funkcije cilja.

Rješenje:

Matematički je model gotovo isti kao i u Primjeru 5. Naznačenim uvjetima dodajemo još jedan. Vrijednost svake pojedine varijable treba biti barem 1 000, što znači da moraju vrijediti nejednakosti:

$$x_1, x_2, x_3, x_4 \geq 1000.$$

Tako dobivamo sljedeći matematički model:

$$\text{maksimizirati } f(x_1, x_2, x_3, x_4) = 1.5 \cdot x_1 + 2.5 \cdot x_2 + 3 \cdot x_3 + 4.5 \cdot x_4$$

pod uvjetima

$$2 \cdot x_1 + 4 \cdot x_2 + 3 \cdot x_3 + 7 \cdot x_4 \leq 100\,000,$$

$$3 \cdot x_1 + 2 \cdot x_2 + 3 \cdot x_3 + 4 \cdot x_4 \leq 50\,000,$$

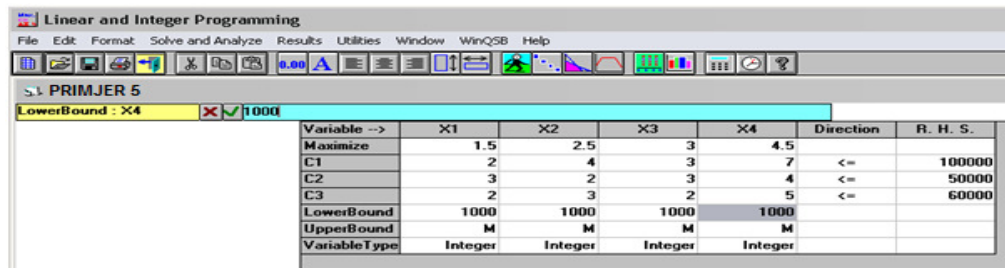
$$2 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + 5 \cdot x_4 \leq 60\,000,$$

$$x_1, x_2, x_3, x_4 \geq 1\,000,$$

$$x_1, x_2, x_3, x_4 \in \mathbf{N}.$$

Koristeći opciju „Load problem“ otvorimo datoteku *Primjer5.lpp*. U pohranjenoj tablici promijenimo donje granice vrijednosti varijabli odlučivanja. Najmanja vrijednost svake spomenute varijable jednaka je 1000, pa u retku *LowerBound* umjesto 0 pišemo 1000. Tako dobivamo tablicu prikazanu na Slici 4.17.

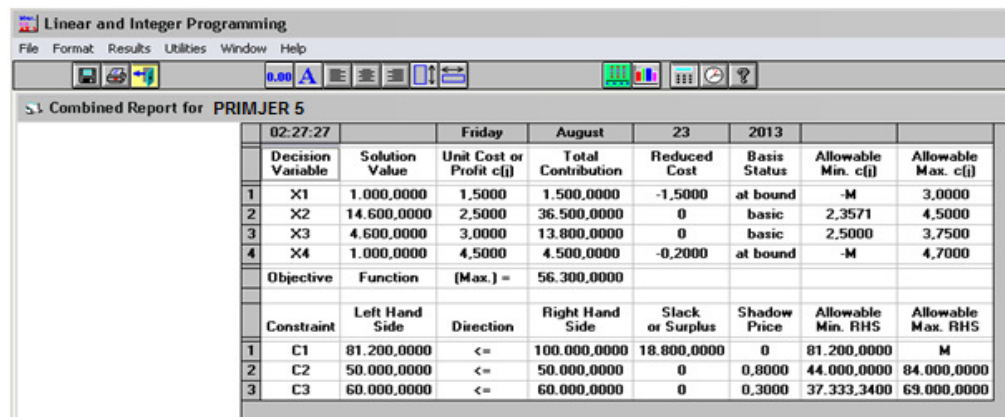
Slika 4.17. Tablica s koeficijentima matematičkoga modela za Primjer 6 (izradio autor)



Variable →	X1	X2	X3	X4	Direction	R. H. S.
Maximize	1.5	2.5	3	4.5		
C1	2	4	3	7	<=	100000
C2	3	2	3	4	<=	50000
C3	2	3	2	5	<=	60000
LowerBound	1000	1000	1000	1000		
UpperBound	M	M	M	M		
VariableType	Integer	Integer	Integer	Integer		

Pripadna tablica s optimalnim vrijednostima prikazana je na Slici 4.18.

Slika 4.18. Tablica s optimalnim vrijednostima za Primjer 6 (izradio autor)



02:27:27		Friday	August	23	2013		
Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	Allowable Min. c(j)	Allowable Max. c(j)
1 X1	1.000,0000	1,5000	1.500,0000	-1,5000	at bound	-M	3,0000
2 X2	14.600,0000	2,5000	36.500,0000	0	basic	2,3571	4,5000
3 X3	4.600,0000	3,0000	13.800,0000	0	basic	2,5000	3,7500
4 X4	1.000,0000	4,5000	4.500,0000	-0,2000	at bound	-M	4,7000
Objective	Function	(Max.) =	56.300,0000				
Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	Allowable Min. RHS	Allowable Max. RHS
1 C1	81.200,0000	<=	100.000,0000	18.800,0000	0	81.200,0000	M
2 C2	50.000,0000	<=	50.000,0000	0	0,8000	44.000,0000	84.000,0000
3 C3	60.000,0000	<=	60.000,0000	0	0,3000	37.333,3400	69.000,0000

Optimalno rješenje ovoga problema je $(x_1^*, x_2^*, x_3^*, x_4^*) = (1\ 000, 14\ 600, 4\ 600, 1\ 000)$.

Optimalna vrijednost funkcije cilja iznosi 56 300. Dakle, treba proizvesti ukupno po 1 000 komada svakoga od klima-uređaja tipa 1 i tipa 4, 14 600 komada klima-uređaja tipa 2 i 4 600 komada klima-uređaja tipa 3. Optimalna ukupna dobit iznosi 56 300 n.j.

Relativne promjene originalnih optimalnih vrijednosti varijabli x_1 i x_4 nema smisla računati jer su te originalne vrijednosti jednake 0. Optimalna vrijednost varijable x_2 smanjila se za $16\ 000 - 14\ 600 = 1\ 400$, odnosno za $\frac{1\ 400}{16\ 000} \cdot 100 = 8.75\ %$.

Optimalna vrijednost varijable x_3 smanjila se za $6\ 000 - 4\ 600 = 1\ 400$, odnosno za $\frac{1\ 400}{6\ 000} \cdot 100 \approx 23.3\ %$. Optimalna vrijednost funkcije cilja smanjila se za $58\ 000 -$

$56\ 300 = 1\ 700$, odnosno za $\frac{1\ 700}{56\ 300} \cdot 100 \approx 3.02\ %$.

4.7. Primjer 7

Riješimo prethodni primjer uz dodatni uvjet da se ukupno godišnje radno vrijeme od $100\ 000 + 50\ 000 + 60\ 000 = 210\ 000$ minuta rasporedi proizvoljno na pojedine operacije. Odredimo relativnu promjenu optimalne vrijednosti svake pojedine varijable (ako je to moguće), kao i relativnu promjenu optimalne vrijednosti funkcije cilja.

Rješenje:

Ukupno vrijeme za provedbu svih triju operacija iznosi

$$2 \cdot x_1 + 4 \cdot x_2 + 3 \cdot x_3 + 7 \cdot x_4 + 3 \cdot x_1 + 2 \cdot x_2 + 3 \cdot x_3 + 4 \cdot x_4 + 2 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + 5 \cdot x_4 = 7 \cdot x_1 + 9 \cdot x_2 + 8 \cdot x_3 + 16 \cdot x_4 \text{ minuta.}$$

To vrijeme ne smije biti strogo veće od 210 000 minuta, pa dobivamo uvjet:

$$7 \cdot x_1 + 9 \cdot x_2 + 8 \cdot x_3 + 16 \cdot x_4 \leq 210\ 000.$$

Stoga je pripadni matematički model:

$$\text{maksimizirati } f(x_1, x_2, x_3, x_4) = 1.5 \cdot x_1 + 2.5 \cdot x_2 + 3 \cdot x_3 + 4.5 \cdot x_4$$

pod uvjetima

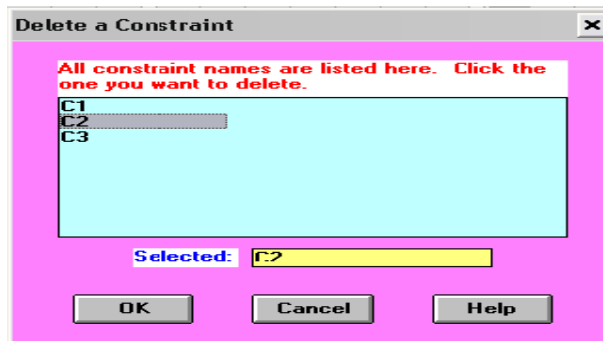
$$7 \cdot x_1 + 9 \cdot x_2 + 8 \cdot x_3 + 16 \cdot x_4 \leq 210\ 000,$$

$$x_1, x_2, x_3, x_4 \geq 1\ 000,$$

$$x_1, x_2, x_3, x_4 \in \mathbf{N}$$

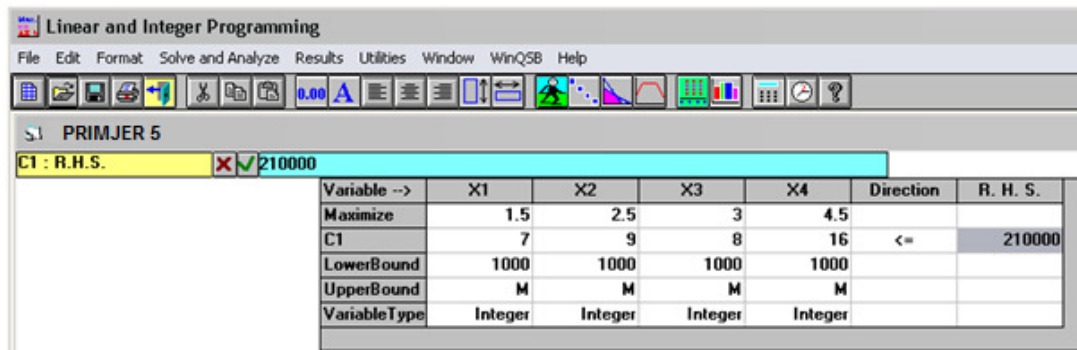
Koristeći opcije iz izbornika *Edit* namijenjen mijenjanju i dodavanju varijabli, odnosno uvjeta, u ovom slučaju potrebno je obrisati uvjete koje smo učitali u prethodnom zadatku i unijeti jedan novi uvjet. Nakon odabira opcije za brisanje uvjeta otvara se izbornik prikazan na Slici 4.19. pomoću kojega se brišu željeni uvjeti.

Slika 4.19. Izbornik za brisanje uvjeta (izradio autor)



Obrišemo uvjete C2 i C3. U redak označen s C1 upišemo redom koeficijente: 7, 9, 8 16, 210000. Time se dobiva tablica prikazana na Slici 4.20.

Slika 4.20. Tablica s koeficijentima matematičkoga modela za Primjer 7 (izradio autor)






Variable -->	X1	X2	X3	X4	Direction	R. H. S.
Maximize	1.5	2.5	3	4.5		
C1	7	9	8	16	<=	210000
LowerBound	1000	1000	1000	1000		
UpperBound	M	M	M	M		
VariableType	Integer	Integer	Integer	Integer		

Pripadna tablica s optimalnim vrijednostima prikazana je na Slici 4.21.

Slika 4.21. Tablica s optimalnim vrijednostima za Primjer 7 (izradio autor)

Linear and Integer Programming

FileFormatResultsUtilitiesWindowHelp



Combined Report for PRIMJER 5

	02:36:56		Friday	August	23	2013		
	Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	Allowable Min. c(j)	Allowable Max. c(j)
1	X1	1,000,0000	1,5000	1,500,0000	-1,1250	at bound	-M	2,6250
2	X2	1,000,0000	2,5000	2,500,0000	-0,8750	at bound	-M	3,3750
3	X3	22,250,0000	3,0000	66,750,0000	0	basic	2,2500	M
4	X4	1,000,0000	4,5000	4,500,0000	-1,5000	at bound	-M	6,0000
	Objective	Function	(Max.) =	75,250,0000				
	Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	Allowable Min. RHS	Allowable Max. RHS
1	C1	210,000,0000	<=	210,000,0000	0	0,3750	40,000,0000	M

Iz dobivene tablice vidimo da je optimalno rješenje ovoga problema $(x_1^*, x_2^*, x_3^*, x_4^*) = (1\ 000, 1\ 000, 22\ 250, 1\ 000)$, dok optimalna vrijednost funkcije cilja iznosi 75 250. Dakle, treba proizvesti po 1 000 komada svakoga od klima-uređaja tipa 1, 2 i 4, te 22 250 komada klima-uređaja tipa 3. Optimalna ukupna dobit iznosi 75 250 n.j.

U odnosu na rješenje Primjera 6., primjećujemo da su optimalne vrijednosti varijabli x_1 i x_4 ostale nepromijenjene (i jednake 1 000), optimalna vrijednost varijable x_2 se smanjila za $14\ 600 - 1\ 000 = 13\ 600$, odnosno za $\frac{13\ 600}{14\ 600} \cdot 100 \approx 93.15\%$, dok se optimalna vrijednost varijable x_3 povećala za $22\ 250 - 4\ 600 = 17\ 650$, odnosno za $\frac{17\ 650}{4\ 600} \cdot 100 \approx 383.7\%$. Pripadna optimalna vrijednost funkcije cilja se povećala za $75\ 250 - 56\ 300 = 18\ 950$, odnosno za $\frac{18\ 950}{56\ 300} \cdot 100 \approx 33.7\%$.

Usporedimo li dobiveno optimalno rješenje s optimalnim rješenjem Primjera 5., relativno povećanje ukupne dobiti iznosilo bi približno 35.7%. Stoga bi tvrtka trebala razmotriti novu razdiobu ukupnoga vremena na pojedine operacije.

4.8. Primjer 8

Lokalni proizvođač domaćega sirupa od višanja proizvodi dvije vrste i prodaje ih na lokalnoj tržnici. Pritom ostvaruje dobit od 5 kn po litri prve vrste i 4 kn po litri druge vrste sirupa od višanja. Prva vrsta domaćega sirupa od višanja zahtijeva 3 kilograma višanja, 4 kilograma šećera i 2 štapića cimeta. Druga vrsta domaćega sirupa od višanja zahtijeva 4 kilograma višanja, 2 kilograma šećera i 1 štapić cimeta. Proizvođač posjeduje 20 kilograma višanja, 18 kilograma šećera i 25 štapića cimeta. Treba naći plan proizvodnje obiju vrsta tako da pripadna dobit bude maksimalna. Proizvedeni obujam svake vrste sirupa mora biti cjelobrojan.

Rješenje:

Neka su x_1 i x_2 redom obujam prve, odnosno druge vrste domaćeg sirupa od višanja. x_1 i x_2 su nenegativni cijeli brojevi, pa vrijedi nejednakost:

$$x_1, x_2 \in \mathbf{N}_0.$$

Proizvodnjom x_1 litara prve vrste sirupa utroši se $3 \cdot x_1$ kg višanja, a proizvodnjom x_2 litara druge vrste sirupa utroši se $4 \cdot x_2$ kg višanja. Stoga je ukupna utrošena masa višanja $3 \cdot x_1 + 4 \cdot x_2$ kg. Budući na raspolaganju imamo 20 kg višanja, mora vrijediti nejednakost:

$$3 \cdot x_1 + 4 \cdot x_2 \leq 20.$$

Proizvodnjom x_1 litara prve vrste sirupa utroši se $4 \cdot x_1$ kg šećera, a proizvodnjom x_2 litara druge vrste sirupa utroši se $2 \cdot x_2$ kg šećera. Stoga je ukupna utrošena masa šećera $4 \cdot x_1 + 2 \cdot x_2$ kg. Budući na raspolaganju imamo 18 kg šećera, mora vrijediti nejednakost:

$$4 \cdot x_1 + 2 \cdot x_2 \leq 18.$$

Proizvodnjom x_1 litara prve vrste sirupa utroši se $2 \cdot x_1$ štapića cimeta, a proizvodnjom x_2 litara druge vrste sirupa utroši se $1 \cdot x_2$ štapića cimeta. Stoga je ukupna utrošena količina cimeta $2 \cdot x_1 + 1 \cdot x_2$ štapića. Budući na raspolaganju imamo 25 štapića cimeta, mora vrijediti nejednakost:

$$2 \cdot x_1 + 1 \cdot x_2 \leq 25.$$

Proizvodnjom x_1 litara prve vrste sirupa ostvari se dobit od $5 \cdot x_1$ kn, a proizvodnjom x_2 litara druge vrste sirupa ostvari se dobit od $4 \cdot x_2$ kn. Stoga je ukupna ostvarena dobit $5 \cdot x_1 + 4 \cdot x_2$ kn i taj izraz treba maksimizirati.

Tako smo dobili sljedeći matematički model:

$$\max f(x_1, x_2) = 5 \cdot x_1 + 4 \cdot x_2$$

pod uvjetima

$$3 \cdot x_1 + 4 \cdot x_2 \leq 20$$

$$4 \cdot x_1 + 2 \cdot x_2 \leq 18$$

$$2 \cdot x_1 + 1 \cdot x_2 \leq 25$$

$$x_1, x_2 \geq 0$$

Ovaj problem riješit ćemo na dva načina: koristeći opciju *Solve the Problem* i grafičku metodu.

Otvorimo novu datoteku, pa u njezine polazne podatke upišimo:

- naziv problema: Primjer 8;
- broj varijabli: 2 (u ovom primjeru su dvije varijable);
- broj uvjeta: 3 (u ovom primjeru su tri uvjeta);

Potom kliknemo na kružice pored sljedećih natpisa:

- *Maximization* (u ovom primjeru se radi o maksimizaciji ukupne dobiti);
- *Nonnegative integer* (u ovom primjeru se radi o nenegativnim cijelim brojevima);
- *Spreadsheet Matrix Form* (ulazni podatci bit će prikazani u matričnom obliku).

Tako dobivamo tablicu prikazanu na Slici 4.22.

Slika 4.22. Unos podataka o matematičkom modelu za Primjer 8 (izradio autor)

LP-ILP Problem Specification

Problem Title: PRIMJER 8

Number of Variables: 2

Number of Constraints: 3

Objective Criterion

☒ Maximization

☐ Minimization

Default Variable Type

☐ Nonnegative continuous

☒ Nonnegative integer

☐ Binary {0,1}

☐ Unsigned/unrestricted

Data Entry Format

☒ Spreadsheet Matrix Form

☐ Normal Model Form

OK Cancel Help

Kliknemo na OK. U dobivenu tablicu je potrebno unijeti koeficijente uz varijable x_1 i x_2 u funkciji cilja i uvjetima. Upisujemo:

- u redak *Maximize*: 5, 4
- u redak *C1*: 3, 4, 20
- u redak *C2*: 4, 2, 18
- u redak *C3*: 2, 1, 25

Tako dobivamo tablicu prikazanu na Slici 4.23.

Slika 4.23. Tablica s koeficijentima matematičkoga modela za Primjer 8 (izradio autor)

Variable →	X1	X2	Direction	R. H. S.
Maximize	5	4		
C1	3	4	<=	20
C2	4	2	<=	18
C3	2	1	<=	25
LowerBound	0	0		
UpperBound	M	M		
VariableType	Integer	Integer		

Pripadna tablica s optimalnim vrijednostima prikazana je na Slici 4.24.

Slika 4.24. Tablica s optimalnim vrijednostima za Primjer 8 (izradio autor)

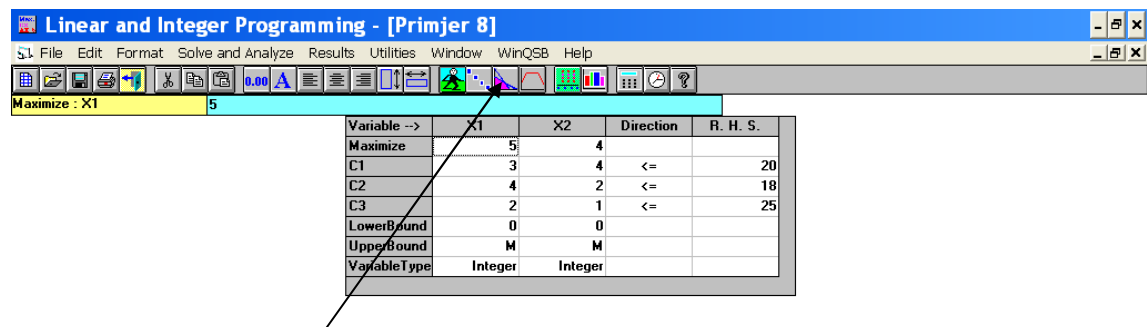
22:59:34 Saturday August 31 2013						
	Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status
1	X1	4,0000	5,0000	20,0000	-3,0000	at bound
2	X2	1,0000	4,0000	4,0000	0	basic
	Objective Function		(Max.) =	24,0000		
	Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price
1	C1	16,0000	<=	20,0000	4,0000	0
2	C2	18,0000	<=	18,0000	0	2,0000
3	C3	9,0000	<=	25,0000	16,0000	0

Iz dobivene tablice vidimo da je optimalno rješenje ovoga problema $(x_1^*, x_2^*) = (4, 1)$, dok optimalna vrijednost funkcije cilja iznosi 24. Dakle, treba proizvesti 4 litre prve

vrste domaćega soka od višanja i 1 litru druge vrste domaćega soka od višanja. Optimalna ukupna dobit iznosi 24 kn.

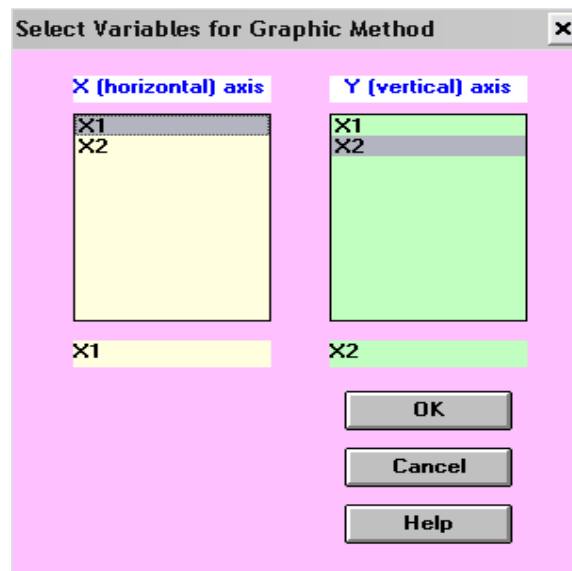
Riješimo ovaj problem i grafičkom metodom. Zatvorimo tablicu sa Slike 4.24., pa kliknemo na ikonicu „grafa“ prikazanu na Slici 4.25.

Slika 4.25. Prikaz ikonice za grafičku metodu rješavanja



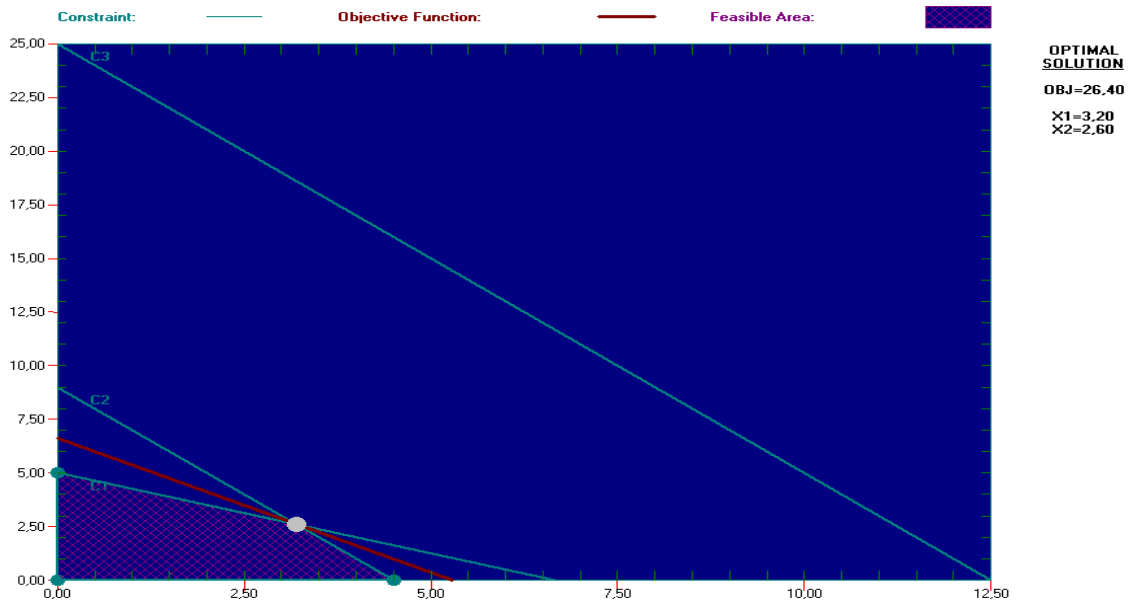
Pojavljuje se izbornik u kojem možemo odabrati koja varijabla će biti na vodoravnoj osi (osi x), a koja na vertikalnoj osi (osi y). Opredijelimo se da varijabla x_1 bude na vodoravnoj, a varijabla x_2 na vertikalnoj osi (vidjeti Sliku 4.26.)

Slika 4.26. Odabir varijabli za grafičku metodu (izradio autor)



Kliknemo na *OK*. Dobivamo grafički prikaz prikazan na Slici 4.27.

Slika 4.27. Rezultat primjene grafičke metode na rješavanje Primjera 8 (izradio autor)



Primjećujemo da u ovom slučaju dobivamo $(x_1^*, x_2^*) = (3.2, 2.6)$ i $f^* = 26.40$, odnosno rezultat različit od rezultata dobivenoga opcijom *Solve the Problem*. Razlog tome jest činjenica da grafička metoda pretpostavlja da su moguće vrijednosti obiju varijabli realni, a ne cijeli brojevi. Stoga će rezultati dobiveni primjenom te metode uvijek biti realni, odnosno decimalni brojevi.

Uočimo da je promjena uvjeta cjelobrojnosti proizvedenoga obujma svake pojedine vrste soka utjecala i na promjenu optimalne dobiti. Ona iznosi 26.40 kn i za $\frac{26.40 - 24}{24} \cdot 100 = 10\%$ je veća u odnosu na optimalnu dobit iz rješenja dobivenoga opcijom *Solve the Problem*. Stoga bi se proizvođaču isplatilo razmotriti opciju proizvodnje obiju vrsta sokova i u pakovanjima od npr. 0.5 l, 0.75 l, 1.5 litara itd.

5. ZAKLJUČAK

Linearno programiranje je zaslužno za dio napretka u prošlom stoljeću koji je omogućio ekonomskoj znanosti rješavanje složenih problema i olakšao donošenje velikih i malih odluka, te ispunjenje ciljeva u praktičnim situacijama.

Primjeri u kojima linearno programiranje može pomoći ljudima su mnogobrojni i raznovrsni. U ovom radu je detaljno riješeno nekoliko njih. Računalni program korišten za rješavanje problemskih zadataka u ovom radu je WinQSB. On pripada u skupinu računalnih programa koji je bolje prilagođen korisnicima svih uzrasta, zanimanja i interesa. Napomenimo da su razvojem novih tehnologija napravljeni i novi računalni programi namijenjeni rješavanju problema linearnoga programiranja.

U primjerima je pokazano kako npr. dodavanje uvjeta o cjelobrojnosti varijable nekim slučajevima može promijeniti optimalno rješenje i optimalnu vrijednost funkcije cilja. To znači da je vrlo bitno precizno definirati tip svake pojedine varijable, ma koliko god takvo definiranje izgledalo beznačajno.

Osim lakšega i bržega rješavanja problema, računalni program omogućuje da se u nekoliko koraka upišu željene promjene u polaznom matematičkom modelu i jednako brzo dobiju nova rješenja. U radu su objašnjeni načini na koje se unose i mijenjaju varijable, uvjeti i sl.

Riješeni primjeri nastoje dati što detaljniji prikaz korištenja programa WinQSB u rješavanju problema linearnoga programiranja vezanima uz probleme planiranja proizvodnje, pa samim time nastoje ukazati mogućim korisnicima na korist toga programa u planiranju njihovoga poslovanja.

6. POPIS KRATICA I AKRONIMA

n.j. = novčanih jedinica

i sl. = i slično

itd. = i tako dalje

7. POPIS LITERATURE I DRUGIH IZVORA PODATAKA

Knjige i udžbenici

- [1] Chiang, C.A.; (1994), Matematičke metode ekonomije, Zagreb, Mate, str. 651. - 665.
- [2] Neralić, L., Šego, B. (2009) Matematika, Zagreb, Element , str. 277. – 314.
- [3] Neralić, L. (2008.) Uvod u matematičko programiranje 1, Zagreb, Element

Internetski izvori

- [4] <http://people.brunel.ac.uk/~mastijb/jeb/or/lp.html> (javno dostupno 12.8.2013.)
- [5] <http://www.eng.buffalo.edu/.../Linear%20programming%20formulation%20example.s.doc> (javno dostupno 12.8.2013.)

Stručni članci

- [6] Jelić, S. (2009) Aproksimacijski algoritmi za optimalno osvjetljenje scene
(URL: [http://www.mathos.unios.hr/~sjelic/papers/Slobodan%20Jelic%20%20Aproksimacijski%20algoritmi%20za%20optimalno%20osvjetljenje%20scene%20\(diplomski%20rad.pdf](http://www.mathos.unios.hr/~sjelic/papers/Slobodan%20Jelic%20%20Aproksimacijski%20algoritmi%20za%20optimalno%20osvjetljenje%20scene%20(diplomski%20rad.pdf)) (javno dostupno 12.8.2013.)
- [7] Losonczi L. (2012) Applications of WinQSB
(URL: <http://www.math.klte.hu/~losi/jegyzet/eco/WinQSBappl.pdf>) (javno dostupno: 12.8.2013.)
- [8] Pavlinović R., Vuković E., Linearno programiranje na primjeru investicija,
(URL: <http://prezi.com/nj79bjavsgpk/linearno-programiranje/>) (javno dostupno: 12.8.2013.)
- [9] Petkoviček D. Linearno programiranje,
(URL: <http://matematika.fkit.hr/staro/izborna/referati/Daniela%20Petkovicsek%20-%20Linearno%20programiranje.pdf>) (javno dostupno 12.8.2013.)
- [10] Schulze M. A.; (1998.); Linear Programming for Optimization;
(URL: <http://www.markschulze.net/LinearProgramming.pdf>) (javno dostupno 12.8.2013.)

8. POPIS SLIKA I TABLICA

Popis slika

Slika 3.1. Pokretanje aplikacijskog modula za linearno i cjelobrojno linearno programiranje..	6
Slika 3.2. Odabir opcije za rješavanje novog zadatka.....	7
Slika 3.3. Tablica za unos podataka o matematičkom modelu.....	7
Slika 3.4. Tablica za upis koeficijenata.....	8
Slika 3.5. Izbornik „Edit“	8
Slika 3.6. Odabir opcije za rješavanje problema klikom na ikonicu	9
Slika 3.7. Odabir opcije „Solve and Analyze“ za rješavanje problema	9
Slika 3.8. Obavijest da je rješenje nađeno.....	9
Slika 3.9. Izlazna tablica.....	10
Slika 4.1. Unos podataka o matematičkom modelu za Primjer 1.....	13
Slika 4.2. Tablica za upis koeficijenata iz Primjera 1	14
Slika 4.3. Unos koeficijenata funkcije cilja i uvjeta u Primjeru 1. Pogreška! Knjižna oznaka nije definirana.	
Slika 4.4. Tablica s optimalnim vrijednostima za Primjer 1	15
Slika 4.5. Tablica s koeficijentima matematičkoga modela za Primjer 2	16
Slika 4.6. Tablica s optimalnim vrijednostima za Primjer 2	17
Slika 4.7. Unos podataka o matematičkom modelu za Primjer 3	21
Slika 4.8. Tablica s koeficijentima matematičkoga modela za Primjer 3	21
Slika 4.9. Tablica s optimalnim vrijednostima za Primjer 3	22
Slika 4.10. Tablica s optimalnim vrijednostima za Primjer 3 uz promjenu tipa varijabli	23
Slika 4.11. Izbornik za dodavanje novoga uvjeta	24
Slika 4.12. Tablica s koeficijentima matematičkoga modela za Primjer 4	25
Slika 4.13. Tablica s optimalnim vrijednostima za Primjer 4	25
Slika 4.14. Unos podataka o matematičkom modelu za Primjer 5	29
Slika 4.15. Tablica s koeficijentima matematičkoga modela za Primjer 5	30
Slika 4.16. Tablica s optimalnim vrijednostima za Primjer 5	30
Slika 4.17. Tablica s koeficijentima matematičkoga modela za Primjer 6	32
Slika 4.18. Tablica s optimalnim vrijednostima za Primjer 6	32
Slika 4.19. Izbornik za brisanje uvjeta	34

Slika 4.20. Tablica s koeficijentima matematičkoga modela za Primjer 7	34
Slika 4.21. Tablica s optimalnim vrijednostima za Primjer 7	35
Slika 4.22. Unos podataka o matematičkom modelu za Primjer 8	38
Slika 4.23. Tablica s koeficijentima matematičkoga modela za Primjer 8	38
Slika 4.24. Tablica s optimalnim vrijednostima za Primjer 7	39
Slika 4.25. Prikaz ikonice za grafičku metodu rješavanja.....	39
Slika 4.26. Odabir varijabli za grafičku metodu.....	40
Slika 4.27. Rezultat primjene grafičke metode na rješavanje Primjera 8	40

Popis tablica

Tablica 3.1. Područje primjene linearnoga programiranja.....	5
Tablica 4.1. Osnovni podatci za Primjer 3.....	18
Tablica 4.2 Osnovni podatci za Primjer 5.....	26