VELEUČILIŠTE U POŽEGI

Josipa Pavić

PRIMJENA RAČUNA VIŠE VARIJABLI NA FUNKCIJE TROŠKOVA, PRIHODA, PROFITA I POTRAŽNJE

Diplomski rad

Lipanj, 2014.

VELEUČILIŠTE U POŽEGI

SPECIJALISTIČKI DIPLOMSKI STUDIJ

TRGOVINSKO POSLOVANJE

PRIMJENA RAČUNA VIŠE VARIJABLI NA FUNKCIJE TROŠKOVA, PRIHODA, PROFITA I POTRAŽNJE

DIPLOMSKI RAD

PREDMET: KVANTITATIVNE METODE U TRGOVINI

Mentor: mr.sc. Bojan Kovačić

Student: Josipa Pavić

Matični broj studenta: 174

Požega, lipanj 2014.

SADRŽAJ:

1. UVOD

Osnovni cilj ovoga rada je pokazati jednostavnost primjene računalnoga programa *Maxima* u rješavanju optimizacije funkcija prihoda, profita, produktivnosti i potražnje primjenom računa više varijabli.

Funkcija troškova je matematička funkcija koja se koristi za predviđanje troškova povezanih s određenom radnjom ili određenom razinom proizvodnje. Tvrtke koriste funkcije troškova kako bi prognozirale troškove povezane s proizvodnjom, tj. kako bi se utvrdilo koje cjenovne strategije treba koristiti kako bi se postigla željena razina profita. [4]

Funkcija prihoda je jednaka umnošku potražnje i cijene robe. [5] Funkcija prihoda predstavlja umnožak količine proizvoda prodanih na tržištu i cijene po kojoj je jedinica proizvoda prodana na tržištu. Količina proizvoda prodanih na tržištu je ustvari funkcija potražnje tog proizvoda. [6]

Funkcija profita je razlika funkcije ukupnog prihoda i ukupnih troškova. [6]

Budući da najveći utjecaj na promjenu količine potražnje nekog dobra ima njegova cijena, funkcija potražnje u užem se smislu definira kao $q = f(p)$. Ona, dakle, iskazuje zavisnost količine potražnje nekog dobra samo o cijeni tog dobra. Naravno, da bi se mogao izmjeriti utjecaj promjene cijene dobra na njegovu potražnju, treba isključiti utjecaj ostalih čimbenika. [7]

2. OSNOVE DIFERENCIJALNOGA RAČUNA FUNKCIJE DVIJU VARIJABLI

2.1. Općenito o parcijalnim derivacijama funkcija dviju varijabli

Neka je $f = f(x, y) : A \to \mathbf{R}$, realna funkcija dviju realnih varijabli čije je prirodno područje definicije skup $A \subseteq \mathbf{R}^2$. Neka je $T = (x_0, y_0) \in A$. Neka je *h prirast nezavisne varijable x* u točki *A*, pri čemu vrijednost nezavisne varijable *y* ostaje nepromijenjena. Neka je $\Delta_x f = f(x_0 + h, y_0) - f(x_0, y_0)$ odgovarajući *prirast funkcije f*. Ako postoji granična vrijednost količnika prirasta funkcije $\Delta_x f$ i prirasta *h* nezavisne varijable *x* kad prirast *h* teži nuli, tada tu graničnu vrijednost nazivamo *parcijalnom derivacijom funkcije f po varijabli x u točki T*. Pišemo:

$$f_x(x_0, y_0) = \lim_{h \to 0} \frac{\Delta_x f}{h} = \lim_{h \to 0} \frac{f(x_0 + h, y_0) - f(x_0, y_0)}{h}.$$

Neka su $f : A \to \mathbf{R}$ realna funkcija dviju realnih varijabli definirana na skupu $A \subseteq \mathrm{R}^2$ i $T = (x_0, y_0) \in A$ bilo koja točka iz skupa *A*. Neka je *h prirast nezavisne varijable y* u promatranoj točki uz nepromijenjenu vrijednost nezavisne varijable *x*, te neka je $\Delta_y f = f(x_0, y_0 + h) - f(x_0, y_0)$ odgovarajući *prirast funkcije f*. Ako postoji granična vrijednost kvocijenta prirasta funkcije $\Delta_y f$ i prirasta *h* nezavisne varijable *y* kad prirast *h* teži nuli, tada tu graničnu vrijednost nazivamo *parcijalnom derivacijom funkcije f po varijabli y u točki T*. Pišemo:

$$f_y(x_0, y_0) = \lim_{h \to 0} \frac{\Delta_y f}{h} = \lim_{h \to 0} \frac{f(x_0, y_0 + h) - f(x_0, y_0)}{h}. \text{ [1]}$$

Ako realna funkcija dviju varijabli ima obje parcijalne derivacije u točki *T*, kažemo da je ta funkcija *diferencijabilna u točki T*. Ako ova tvrdnja vrijedi za bilo koju točku *T* iz skupa *A*, kažemo da je funkcija *f diferencijabilna* (na cijeloj svojoj domeni).

Parcijalne derivacije se u neku ruku mogu shvatiti kao dvije „obične“ derivacije. Naime, prigodom deriviranja funkcije više varijabli po točno jednoj od tih varijabli sve ostale varijable smatramo realnim konstantama, pa se deriviranje u tom slučaju zapravo svodi na deriviranje realne funkcije jedne realne varijable prema poznatim pravilima za deriviranje takvih funkcija.

Kažemo da funkcija f ima *lokalni maksimum u točki* $M = (x_M, y_M)$ ako postoji otvoreni krug K koji sadrži točku M i takav da za svaku točku T iz kruga K vrijedi: $f(T) \leq f(M)$. Ako znak $\leq$ zamijenimo znakom $<$, govorimo o *strogom lokalnom maksimumu*. Kažemo da funkcija f ima *lokalni minimum u točki* $m = (x_m, y_m)$ ako postoji otvoreni krug K koji sadrži točku m i takav da za svaku točku T iz kruga K vrijedi: $f(T) \geq f(m)$. Ako znak $\geq$ zamijenimo znakom $>$, govorimo o *strogom lokalnom minimumu*.

Lokalni maksimum i lokalni minimum jednim imenom nazivamo *lokalni ekstremi* funkcije f.

2.2. Algoritam za određivanje lokalnih ekstrema funkcije dviju varijabli

Algoritam za određivanje lokalnih ekstrema funkcije dviju varijabli glasi:

1. Nađu se stacionarne točke funkcije $z = f(x, y)$, tj. svi uređeni parovi $T = (x_T, y_T) \in A$ za koje su vrijednosti obiju parcijalnih derivacija funkcije f jednake nuli, odnosno za koje vrijedi $f_x(x_0, y_0) = 0$ i $f_y(x_0, y_0) = 0$.

2. Nađu se parcijalne derivacije 2. reda z_{xx}, z_{xy}, z_{yy}. Oznaka z_{xx} znači da parcijalnu derivaciju funkcije z po varijabli x parcijalno deriviramo po varijabli x. Oznaka z_{xy} znači da parcijalnu derivaciju funkcije z po varijabli x parcijalno deriviramo po varijabli y. Oznaka z_{yy} znači da parcijalnu derivaciju funkcije z po varijabli y parcijalno deriviramo po varijabli y. Mi ćemo promatrati funkcije z koje imaju svojstvo $z_{xy} = z_{yx}$

3. Izračuna se vrijednost parcijalnih derivacija

$$h_{11} = z_{xx}(x_T, y_T),\ h_{11} = h_{21} = z_{xy}(x_T, y_T) \text{ i } h_{22} = z_{yy}(x_T, y_T),$$

u svakoj stacionarnoj točki $T = (x_T, y_T)$

4. Formira se *Hesseova* matrica:

$$H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{12} & h_{22} \end{bmatrix}.$$

i računa vrijednost njene determinante $\det(H) = h_{11} \cdot h_{22} - h_{12}^2$.

a) Ako je det(H) > 0, onda funkcija f u promatranoj stacionarnoj točki ima strogi lokalni ekstrem, i to: minimum ako je $h_{11} > 0$, odnosno maksimum ako je $h_{11} < 0$.

b) Ako je det(H) < 0, onda funkcija f u promatranoj stacionarnoj točki nema lokalni ekstrem. U tom slučaju kažemo da je *T sedlasta točka* grafa funkcije f.

c) Ako je det(H) = 0, onda za odluku o postojanju ili nepostojanju lokalnog ekstrema u promatranoj točki valja izvršiti dodatna ispitivanja koja se provode pomoću parcijalnih derivacija višeg reda.

Korake 3. i 4. treba ponoviti zasebno za svaku stacionarnu točku. [5]

3. RAČUNALNI PROGRAM *MAXIMA*

3.1. Općenito o računalnom programu *Maxima*

Maxima je računalni program za algebarske operacije sa simboličkim i numeričkim izrazima. U te operacije ubrajaju se i deriviranje, integriranje, razvoj u Taylorov red, Laplaceova transformacija itd. Pomoću toga programa mogu se rješavati različite algebarske i nealgebarske jednadžbe, obične diferencijalne jednadžbe, sustavi linearnih jednadžbi itd. Rezultati tih operacija su vrlo precizni, a ukupan broj znamenaka u njihovu prikazu ovisi o odabranom obliku zapisa (s točnošću od 10^{-4} („kratak oblik“) ili s točnošću od 10^{-12} („dugačak oblik“)).

Maxima također može izvršavati algebarske operacije s funkcijama više varijabli. Njezin izvorni kod je prilagođen i čitljiv u brojnim operacijskim sustavima, uključujući *Windows*, *Linux* i *MacOS X*. Izvorni kodovi za sve sustave i kompilirane izvršne datoteke za *Windows* i *Linux* dostupni su na „*SourceForge file manager*“. [2]

3.2. Izbornik programa *Maxima*

Glavni izbornik programa *Maxima* sastoji se od 11 izbornika (*File, Edit, Cell, Maxima, Equations, Algebra, Calculus, Simplify, Plot, Numeric, Help*).

U izborniku *File* najvažnije opcije su *New* za otvaranje nove datoteke, *Open* za otvaranje postojeće datoteke, *Save* za pohranu postojeće datoteke i *Export* za „izvoz“ podataka u HTML ili TeX obliku.

U izborniku *Edit* najvažnije opcije su *Undo* za povratak na prethodni korak, *Cut* za „izrezivanje“ odabranih podataka (odnosno njihovo brisanje iz komandnoga prozora), *Copy* za kopiranje odabranih podataka, *Paste* za „lijepljenje“ kopiranih podataka, *Select All* za označavanje svih upisanih podataka, *Zoom In* za povećavanje veličine fonta upisanih podataka, *Zoom Out* za smanjivanje veličine fonta upisanih podataka i *Full Screen* za prikazivanje izbornika programa na cijelom ekranu računala.

U izborniku *Cell* najvažnije opcije su *Insert Input Cell* za umetanje nove ćelije, *Insert Text Cell* za umetanje ćelije za upisivanje teksta, *Insert Title Cell* za umetanje naslova ćelije, *Insert Page Break* za umetanje prelamanja stranice i *Insert Image* za umetanje slike.

U izborniku *Maxima* najvažnije opcije su *Interrupt* za prekidanje rada programa, *Restart Maxima* za ponovno pokretanje programa, *Clear Memory* za brisanje sadržaja memorije, *Show Functions* za prikazivanje prethodno upisanih funkcija, *Show definition* za prikazivanje definicije odabrane funkcije, *Show Variables* za prikazivanje prethodno upisanih varijabli, *Delete Function* za brisanje upisanih funkcija i *Delete Variable* za brisanje upisanih varijabli.

U izborniku *Equations* najvažnije opcije su *Solve* za rješavanje algebarske ili nealgebarske jednadžbe, *Solve Linear System* za rješavanje sustava linearnih jednadžbi, *Solve Algebraic System* za rješavanje sustava algebarskih jednandžbi i *Eliminate Variable* za izražavanje odabrane nepoznanice sustava jednadžbi pomoću ostalih nepoznanica i parametara u sustavu.

U izborniku *Algebra* najvažnije opcije su *Generate Matrix* za generiranje matrice, *Generate Matrix from Expression* za generiranje matrice iz izraza, *Enter Matrix* za upisivanje matrice, *Invert Matrix* za invertiranje matrice, *Determinant* za izračunavanje determinante kvadratne realne matrice i *Transpose Matrix* za transponiranje matrice.

U izborniku *Calculus* najvažnije opcije su *Integrate* za integriranje, *Differentiate* za parcijalno i „obično“ deriviranje, *Find Minimum* za izračunavanje minimuma zadanog izraza i *Calculate Sum* za izračunavanje zbroja određenog izraza.

U izborniku *Simplify* najvažnije opcije su *Simplify Expression* za pojednostavljivanje izraza koji sadrži simboličke objekte (brojeve i varijable), *Expand Expression* za ispisivanje proširenoga zapisa izraza koji sadrži simboličke objekte, *Expand Logarithms* za ispisivanje proširenoga zapisa simboličkih logaritama i *Substitute* za zamjenu varijable u simboličkom izrazu novom varijablom ili simboličkim izrazom.

U izborniku *Plot* najvažnije opcije su *Plot 2d* za crtanje dvodimenzionalnih grafova i *Plot 3d* za crtanje trodimenzionalnih grafova.

U izborniku *Numeric* najvažnija opcija je *To Float* za pretvaranje simboličkih realnih ili kompleksnih brojeva u „obične“ realne ili kompleksne brojeve.

U izborniku *Help* najvažnije opcije su *Maxima Help* za upoznavanje sa radom programa, *Example* za prikazivanje primjera rada određene funkcije i *Apropos* za prikazivanje svih funkcija koje su slične zadanoj funkciji.

Alatne trake sa kontrolama za otvaranje novih prozora sastoji se od 14 sličica (*New document, Open document, Save document, Print document, Configure*

wxMaxima, Cut selection, Copy selection, Paste from clipboard, Find and replace, Interrupt current computation, Start animation, Stop animation, Show Maxima help) (vidjeti Sliku 3.1.).

Opcija *New Document* omogućuje otvaranje nove datoteke.

Opcija *Open document* omogućuje otvaranje postojeće datoteke pohranjene na čvrsti ili prijenosni disk.

Opcija *Save document* omogućuje pohranu postojeće datoteke na čvrsti ili prijenosni disk.

Opcija *Print document* omogućuje ispisivanje datoteke na pisač.

Opcija *Configure wxMaxima* omogućuje podešavanje osnovnih postavki programa *Maxima* (jezik, font, itd.)

Opcija *Cut selection* omogućuje „izrezivanje“ odabranih podataka, odnosno njihovo brisanje iz komandnoga prozora.

Opcija *Copy selection* omogućuje kopiranje odabranih podataka.

Opcija *Paste from clipboard* omogućuje „lijepljenje“ podataka koji se kopiranjem smještavaju u međuspremnik.

Opcija *Find and replace* omogućuje pronalazak teksta i zamjenu s drugim odabranim tekstom.

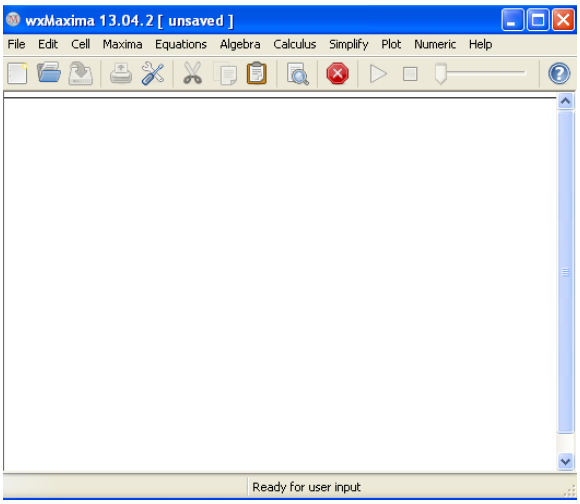
Opcija *Interrupt current computation* omogućuje prekidanje tekućeg računanja.

Opcija *Start animation* omogućuje pokretanje ranije pohranjene animacije.

Opcija *Stop animation* omogućuje zaustavljanje pokrenute animacije.

Opcija *Show Maxima help* omogućuje upoznavanje sa radom programa.

Slika 3.1. Izbornik programa *Maxima*



3.3. Pregled funkcija računalnoga programa *Maxima* korištenih pri rješavanju zadataka

Funkcije korištene u rješavanju zadataka u primjerima su `expand`, `diff`, `solve`, `hessian`, `determinant`, `float`, `max`, `abs` i `at`. [3]

Funkcija `expand` omogućava ispis proširenoga zapisa simboličkih objekata (funkcija, algebarskih izraza, simboličkih brojeva itd.). Stoga je jedini argument ove funkcije neki simbolički objekt.

Funkcija `diff` vraća derivaciju funkcije. Kao argument ove funkcije nužno se navodi funkcija koju treba derivirati. Kao ostali argumenti mogu se navesti broj derivacije, varijabla po kojoj se derivira (u slučaju funkcije više varijabli) itd.

Funkcija `solve` vraća rješenje algebarske ili nealgebarske jednadžbe. Kao argumenti ove funkcije navode se algebarski izraz koji zadaje jednadžbu, te varijabla koja označava nepoznanicu u jednadžbi.

Funkcija `hessian` izračunava Hesseovu matricu funkcije dviju varijabli. Argumenti te funkcije su realna funkcija (barem) dviju realnih varijabli, nazivi varijabli i rangovi varijabli. Rangirati varijable znači navesti koja varijabla je prva, koja druga itd. (npr. ako je $z = f(x, y)$, onda je prva varijabla x, a druga varijabla y).

Funkcija `determinant` izračunava determinantu kvadratne realne matrice.

Funkcija `float` pretvara simboličke realne ili kompleksne brojeve u „obične“ realne ili kompleksne brojeve s kojima se dalje može računati kao s brojevima, a ne kao sa simboličkim objektima. Jedini argument te funkcije je simbolički broj kojega treba „pretvoriti“ u „običan“ realan broj.

Funkcija `max` vraća najveći element konačnoga podskupa skupa realnih brojeva. Jedini argument te funkcije je konačni podskup skupa realnih brojeva.

Funkcija `abs` vraća apsolutnu vrijednost realnoga ili kompleksnoga broja. Jedini argument te funkcije je realan, odnosno kompleksan broj.

Funkcija `at` namijenjena je uvrštavanju simboličkih vrijednosti (brojeva) u simbolički izraz, odnosno računanju vrijednosti funkcije u točki koja pripada njezinu prirodnu području definicije.

4. PRIMJENA RAČUNA VIŠE VARIJABLI NA FUNKCIJE TROŠKOVA, PRIHODA, PROFITA I POTRAŽNJE

4.1. Primjer 1.

Prodavaonica mješovite robe „*Sve za malo kuna*“ u Frkljevcima nudi dvije vrste soka od jabuke: „*Jabučko*“ čija je nabavna jedinična cijena 40 n.j. i „*Applejuice*“ čija je nabavna jedinična cijena 50 n.j. Na temelju analize tržišta procijenjeno je da će prodaja soka „*Jabučko*“ po jediničnoj cijeni od x n.j. i soka „*Applejuice*“ čija je jedinična cijena y n.j. rezultirati dnevnom prodajom od približno $72 - 4.5 \cdot x + 3 \cdot y$ bočica „*Jabučka*“ i $86 + 7 \cdot x - 7.3 \cdot y$ bočica soka „*Applejuice*“. Treba odrediti jedinične cijene obiju vrsta soka tako da dobit od prodaje sokova bude maksimalna. Pritom optimalne vrijednosti varijabli x i y moraju biti nenegativni cijeli brojevi. [9]

Ukupna dnevna dobit dobiva se zbrajanjem dobiti od prodaje soka „*Jabučko*“ i dobiti od prodaje soka „*Applejuice*“. Dnevna dobit od prodaje svakoga pojedinoga soka izračunava se množenjem razlike prodajne i nabavne cijene toga soka s ukupnom dnevnom prodanom količinom toga soka.

Neka je $C_1(x, y)$ funkcija koja označava razliku prodajne i nabavne cijene soka „*Jabučko*“. Očito je:

$$C_1(x, y) = x - 40.$$

Neka su $P_1(x, y)$ i $Q_1(x, y)$ funkcija koja označava ukupnu dnevnu dobit nastalu prodajom soka „*Jabučko*“, odnosno funkcija koja označava ukupnu dnevnu prodanu količinu toga soka. Prema podatcima iz zadatka je:

$$Q_1(x, y) = 72 - 4.5 \cdot x + 3 \cdot y,$$

pa ukupna dnevna dobit iznosi:

$$P_1(x, y) = C_1(x) \cdot Q_1(x, y) = (x - 40) \cdot (72 - 4.5 \cdot x + 3 \cdot y).$$

Ove funkcije upisuju se u program *Maxima*. Nakon pokretanja programa u prvi redak se upisuje funkcija koja označava razliku prodajne i nabavne cijene soka „*Jabučko*“:

```
C1(x,y):=x-40;
```

Pritisne se tipka *Enter*. Potom se u novi redak upisuje funkcija ukupne dnevne prodane količine soka „*Jabučko*“:

```
Q1(x,y):=72-4.5*x+3*y;
```

Pritisne se tipka *Enter*. U novi redak upisuje se funkcija ukupne dnevne dobiti nastale prodajom soka „ *Jabučko*“:

```
P1(x,y):=C1(x,y)*Q1(x,y);
```

Za ispis propisa funkcije $P_1(x, y)$ koristi se funkcija `expand`. U novi redak se utipka:

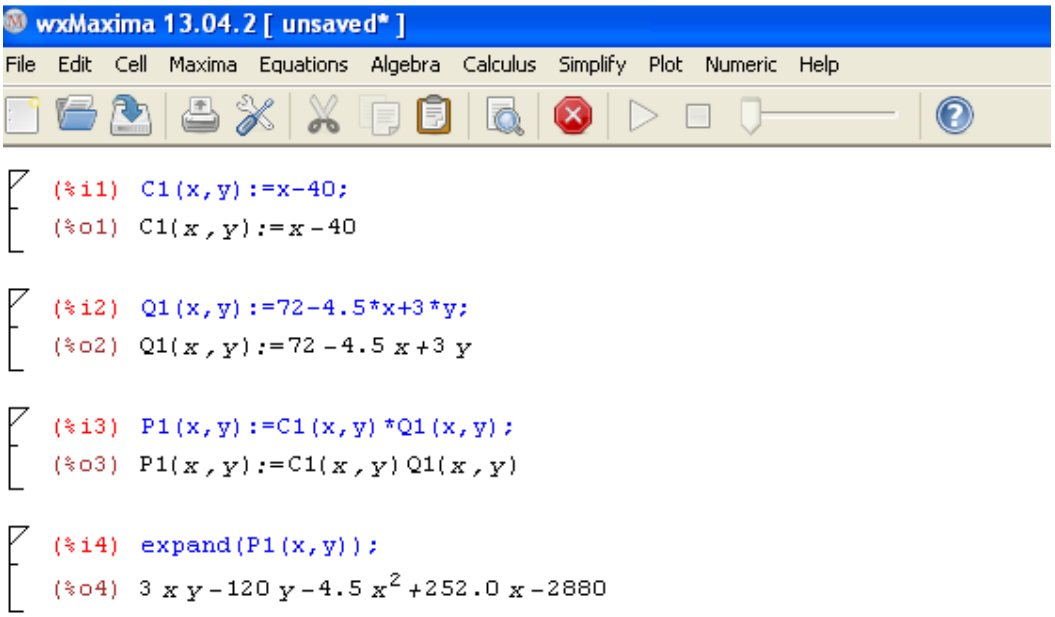
```
expand(P1(x,y));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$3xy-120y-4.5x^2+252.0x-2880$$

(vidjeti Sliku 4.1.).

Slika 4.1. Zadavanje funkcija C_1, Q_1 i P_1 u Primjeru 1.



Dakle,

$$P_1(x, y) = 3 \cdot x \cdot y - 120 \cdot y - 4.5 \cdot x^2 + 252 \cdot x - 2880.$$

Analogno se postupi i u slučaju soka „*Applejuice*“. Neka je $C_2(x,y)$ funkcija koja označava razliku prodajne i nabavne cijene soka „*Applejuice*“. Očito je:

$$C_2(y) = y - 50.$$

Neka su $P_2(x, y)$ i $Q_2(x, y)$ funkcija koja označava ukupnu dnevnu dobit nastalu prodajom soka „*Applejuice*“, odnosno funkcija koja označava ukupnu dnevnu prodanu količinu toga soka. Prema podatcima iz zadatka je

$$Q_2(x, y) = 86 + 7 \cdot x - 7.3 \cdot y,$$

pa ukupna dnevna dobit iznosi:

$$P_2(x, y) = C_2(y) \cdot Q_2(x, y) = (y - 50) \cdot (86 + 7 \cdot x - 7.3 \cdot y).$$

Ove funkcije upisuju se u program *Maxima*. U novi redak se upisuje funkcija razlike prodajne i nabavne cijene $C_2(x,y)$:

```
C2(x,y):=y-50;
```

Pritisne se tipka *Enter*. Potom se u novi redak upisuje funkcija ukupne dnevne prodane količine $Q_2(x, y)$:

```
Q2(x,y):=86+7*x-7.3*y;
```

Pritisne se tipka *Enter*. U novi redak upisuje se funkcija ukupne dnevne dobiti $P_2(x, y)$:

```
P2(x,y):=C2(x,y)*Q2(x,y);
```

Za ispis propisa funkcije $P_2(x, y)$ u novi redak se utipka:

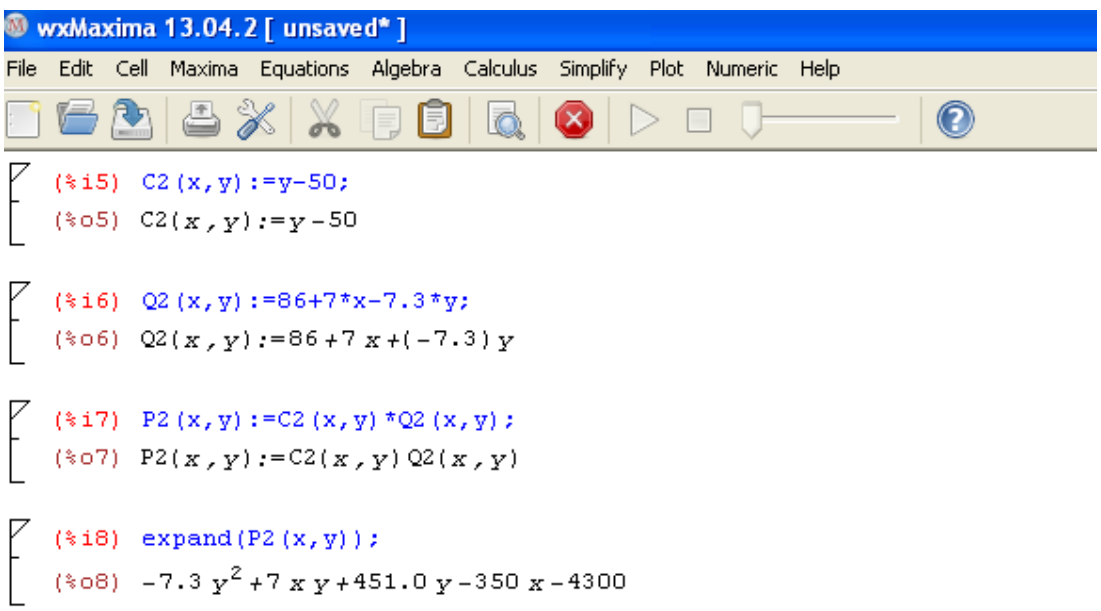
```
expand(P2(x,y));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$-7.3y^2+7xy+451.0y-350x-4300$$

(vidjeti Sliku 4.2.).

Slika 4.2. Zadavanje funkcija C_2, Q_2 i P_2 u Primjeru 1.



Dakle,

$$P_2(x, y) = -7.3 \cdot y^2 + 7 \cdot x \cdot y + 451 \cdot y - 350 \cdot x - 4300.$$

Neka je $P(x, y)$ funkcija ukupne dobiti nastale prodajom obiju navedenih vrsta sokova. Tada je:

$$P(x, y) = P_1(x, y) + P_2(x, y).$$

U novi redak programa *Maxima* utipka se:

```
P(x,y):=P1(x,y)+P2(x,y);
```

Pritisne se *Enter*. Za ispis propisa funkcije *P* u novi redak se utipka:

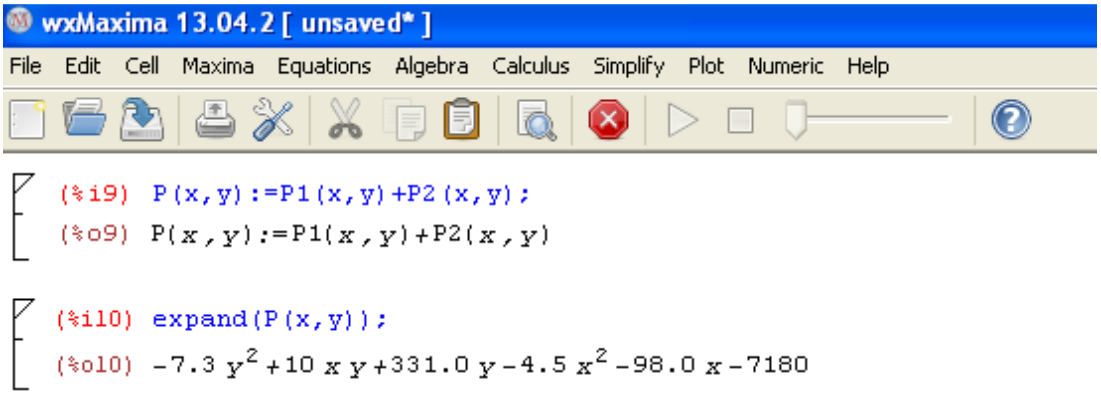
```
expand(P(x,y));
```

Pritisne se *Enter*. *Maxima* će ispisati:

$$-7.3y^2+10xy+331.0y-4.5x^2-98.0x-7180$$

(vidjeti Sliku 4.3.).

Slika 4.3. Određivanje funkcije *P* u Primjeru 1.



Dakle,

$$P(x, y) = -4.5 \cdot x^2 - 7.3 \cdot y^2 + 10 \cdot x \cdot y - 98 \cdot x + 331 \cdot y - 7180.$$

Da bi se odredili lokalni ekstremi funkcije *P*, najprije je potrebno odrediti obje parcijalne derivacije te funkcije. Neka su P_x i P_y redom parcijalna derivacija funkcije *P* po varijabli *x*, odnosno varijabli *y*.

Najprije se određuje P_x. U novi redak programa *Maxima* utipka se:

```
Px(x,y):=diff(P(x,y),x);
```

Pritisne se tipka *Enter*. Za ispis propisa funkcije P_x u novi redak se utipka:

```
expand(Px(x,y));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$10y-9.0x-98.0$$

Dakle,

$$P_x(x, y) = -9 \cdot x + 10 \cdot y - 98.$$

U nastavku se određuje P_y. U novi redak programa *Maxima* se utipka:

```
Py(x,y):=diff(P(x,y),y);
```

Pritisne se tipka *Enter*. Za ispis propisa funkcije P_y u novi redak se utipka:

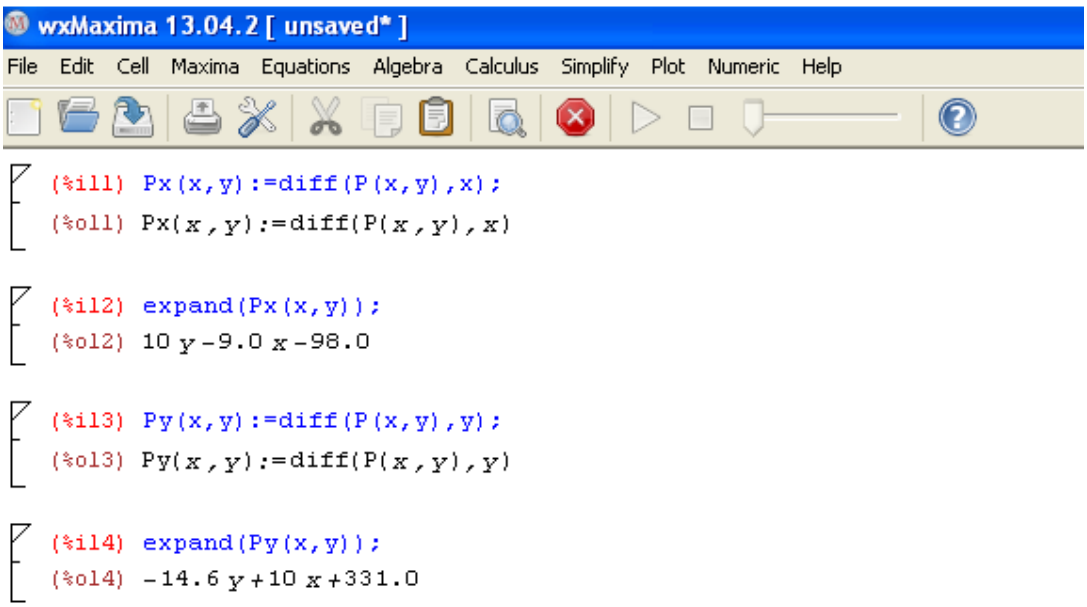
```
expand(Py(x,y))
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
-14.6y+10x+331.0
```

(vidjeti Sliku 4.4.).

Slika 4.4. Određivanje funkcija P_x i P_y u Primjeru 1.



Dakle,

$$P_y(x, y) = 10 \cdot x - 14.6 \cdot y + 331.$$

Stacionarne točke su rješenja sustava jednadžbi:

$$P_x(x, y) = 0$$

$$P_y(x, y) = 0.$$

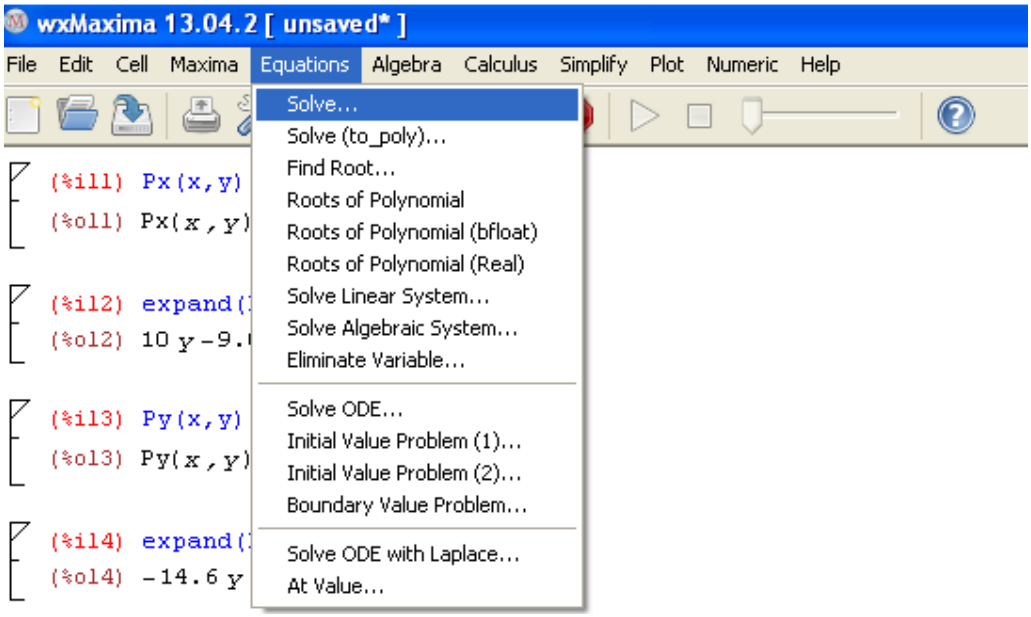
U ovom slučaju to je sustav:

$$-9 \cdot x + 10 \cdot y - 98 = 0$$

$$10 \cdot x - 14.6 \cdot y + 331 = 0.$$

U izborniku se odabere opcija *Equations* i njezina podopcija *Solve* (vidjeti Sliku 4.5.).

Slika 4.5. Opcija *Equations* i njezina podopcija *Solve*



U pravokutnik pored natpisa *Equation*(*s*) u prozorčiću *Solve* upisuje se:

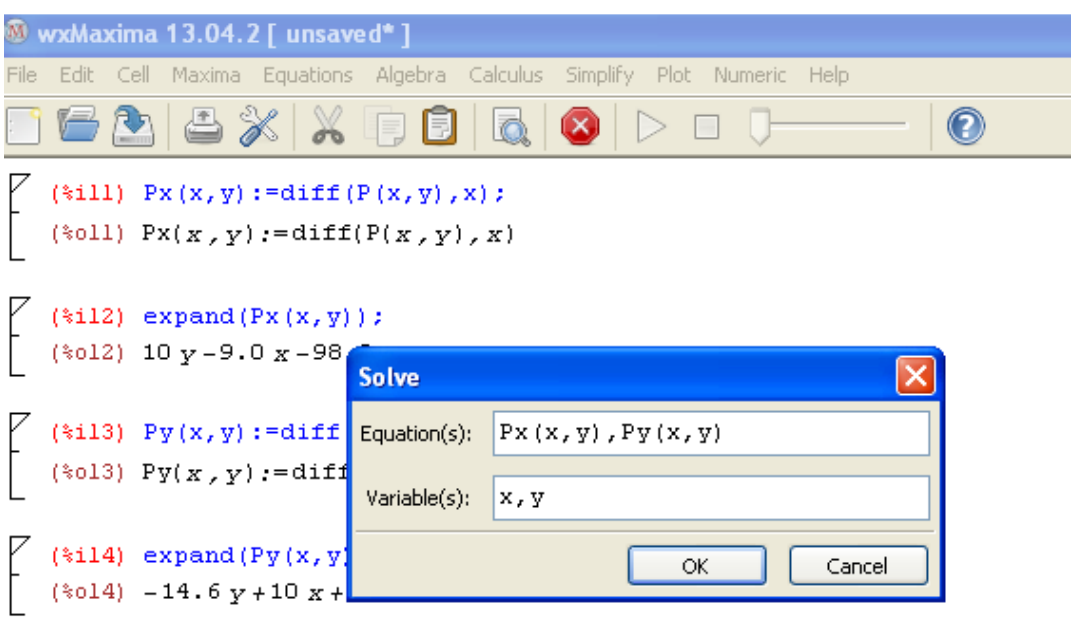
`Px(x,y),Py(x,y)`

U pravokutnik pored natpisa *Variable*(*s*) u prozorčiću *Solve* upisuje se:

`x,y`

(vidjeti Sliku 4.6.).

Slika 4.6. Upisivanje u podopciju *Solve* u Primjeru 1.



Klikne se na OK. *Maxima* će ispisati:

```
rat: replaced -4.5 by -9/2 = -4.5
rat: replaced -4.5 by -9/2 = -4.5
rat: replaced -7.3 by -73/10 = -7.3
rat: replaced -7.3 by -73/10 = -7.3
[[x=9396/157,y=9995/157]]
```

Dakle, stacionarna točka je $T = \left(\frac{9396}{157}, \frac{9995}{157}\right)$.

Treba provjeriti je li točka *T* ujedno i točka lokalnoga maksimuma funkcije *P*. U tu se svrhu računa pripadna *Hesseova* matrica. U novi redak programa *Maxima* utipka se:

```
H:hessian(P(x,y),[x,y]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$\begin{bmatrix} -9.0 & 10 \\ 10 & -14.6 \end{bmatrix}.$$

Dakle,

$$H = \begin{bmatrix} -9 & 10 \\ 10 & -14.6 \end{bmatrix}.$$

Potom se odredi determinanta matrice *H*. U novi redak programa *Maxima* utipka se:

```
determinant(H);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
31.40000000000001
```

Zaključuje se da je $H_{11} = -9 < 0$ i $\det(H) = 31.4 > 0$, pa je *T* točka lokalnoga maksimuma funkcije *P*.

Koordinate točke *T* treba zapisati kao decimalne brojeve. U novi redak programa *Maxima* utipka se:

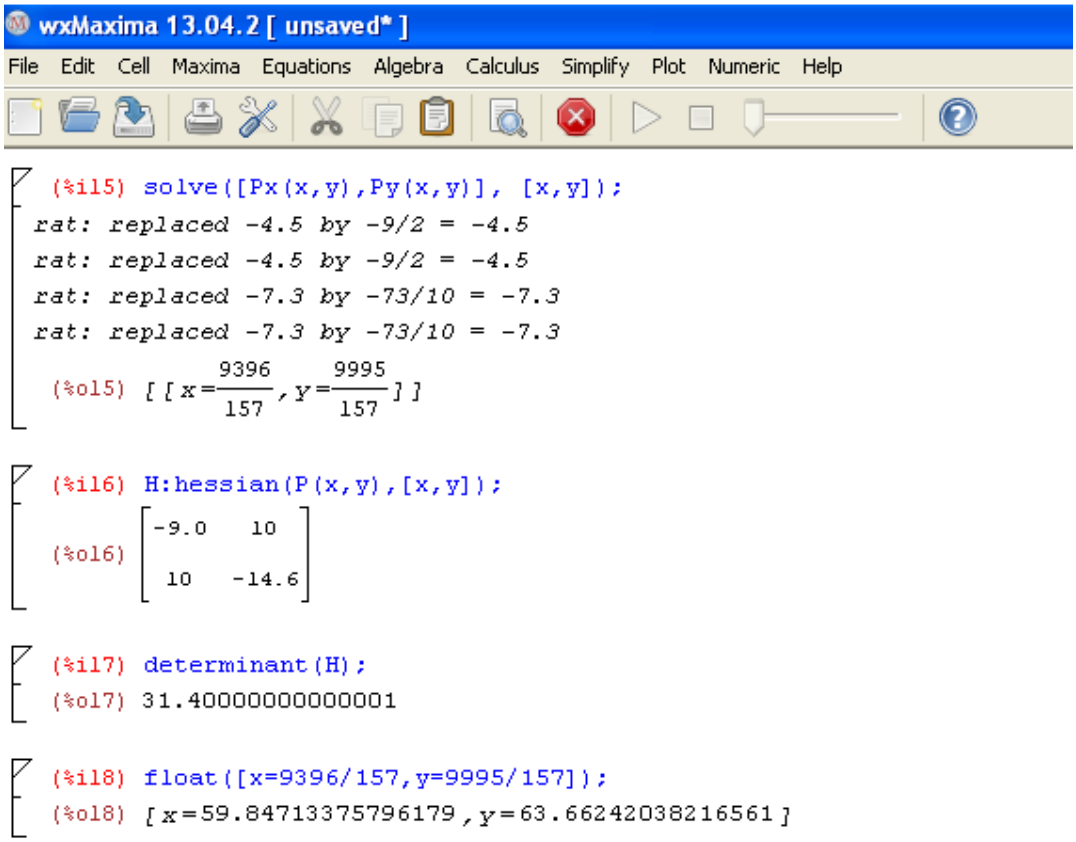
```
float([x=9396/157,y=9995/157]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
[x=59.84713375796179,y=63.66242038216561]
```

(vidjeti Sliku 4.7.).

Slika 4.7. Izračun stacionarne točke i pripadne Hesseove matrice u Primjeru 1.



Budući da optimalne vrijednosti trebaju biti nenegativni cijeli brojevi, mogući su sljedeći slučajevi:

1.) $x^* = 59$, $y^* = 63$;

2.) $x^* = 59$, $y^* = 64$;

3.) $x^* = 60$, $y^* = 63$;

4.) $x^* = 60$, $y^* = 64$.

Optimalna dnevna dobit jednaka je najvećoj od četiriju vrijednosti: $P(59, 63)$, $P(59, 64)$, $P(60, 63)$ i $P(60, 64)$. U novi redak programa *Maxima* utipka se:

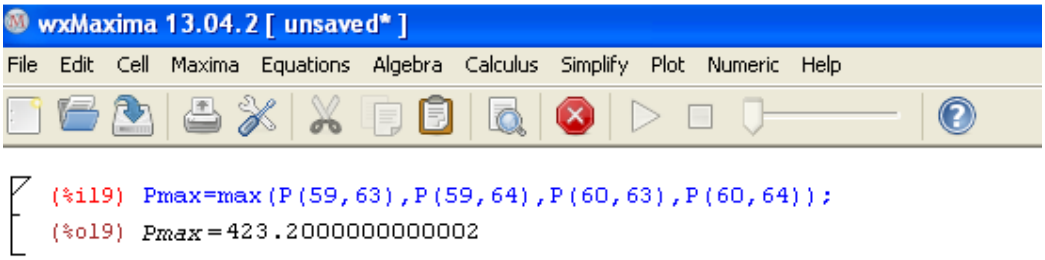
```
Pmax=max(P(59,63),P(59,64),P(60,63),P(60,64));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
Pmax=423.2000000000002
```

(vidjeti Sliku 4.8.).

Slika 4.8. Izračun optimalne dnevne dobiti u Primjeru 1.



Dakle, maksimalna dnevna dobit iznosi 423.2 n.j. Lako se provjeri da je $P(60, 64) = 432.2$, pa se zaključuje da se maksimalna dnevna dobit može ostvariti prodajom sokova „*Jabučko*“ po jediničnoj cijeni od 60 n.j. i sokova „*Applejuice*“ po jediničnoj cijeni od 64 n.j.

4.2. Primjer 2.

Tvrtka „*Krkošija d.d.*“ iz Magadenovca proizvodi vino na dvije lokacije: Malinovac i Šljivoševci. U Malinovcu se proizvodi vino „*Blatina*“ čija je jedinična cijena 28 n.j., a troškovi proizvodnje $0.17 \cdot x^2 + 2 \cdot x + 740$, gdje je x obujam vina (iskazan u litrama). U Šljivoševcima se proizvodi vino „*Crljenak*“ čija je jedinična cijena 28 n.j., a troškovi proizvodnje $0.08 \cdot y^2 + 4 \cdot y + 523$, gdje je y obujam vina (iskazan u litrama). Treba odrediti količine vina koje treba proizvesti na svakoj lokaciji tako da ukupna dobit od prodaje obiju vrsta vina bude maksimalna. Pritom optimalne vrijednosti varijabli x i y moraju biti nenegativni cijeli brojevi. [10]

Ukupna dobit dobiva se oduzimanjem ukupnih troškova proizvodnje obiju vrsta vina od prihoda nastaloga prodajom obiju vrsta vina. Ukupni troškovi proizvodnje obiju vrsta vina jednaki su zbroju troškova proizvodnje vina „*Blatina*“ i troškova proizvodnje vina „*Crljenak*“.

Neka je $C_1(x, y)$ funkcija koja označava troškove proizvodnje vina „*Blatina*“. Očito je:

$$C_1(x, y) = 0.17 \cdot x^2 + 2 \cdot x + 740.$$

Neka je $C_2(x, y)$ funkcija koja označava troškove proizvodnje vina „*Crljenak*“. Očito je:

$$C_2(x, y) = 0.08 \cdot y^2 + 4 \cdot y + 523$$

pa ukupni troškovi proizvodnje obiju vrsta vina iznose:

$$C(x, y) = C_1(x,y) + C_2(x, y).$$

Ove funkcije upisuju se u program *Maxima*. Nakon pokretanja programa u prvi redak se upisuje funkcija koja označava troškove proizvodnje vina „*Blatina*“ :

```
C1(x,y):=0.17*x^2+2*x+740;
```

Pritisne se tipka *Enter*. Potom se u novi redak upisuje funkcija koja označava troškove proizvodnje vina „*Crljenak*“:

```
C2(x,y):=0.08*y^2+4*y+523;
```

Pritisne se tipka *Enter*. U novi redak upisuje se funkcija ukupnih troškova proizvodnje obiju vrsta vina:

```
C(x,y):=C1(x,y)+C2(x,y);
```

Za ispis propisa funkcije $C(x, y)$ u novi redak se utipka:

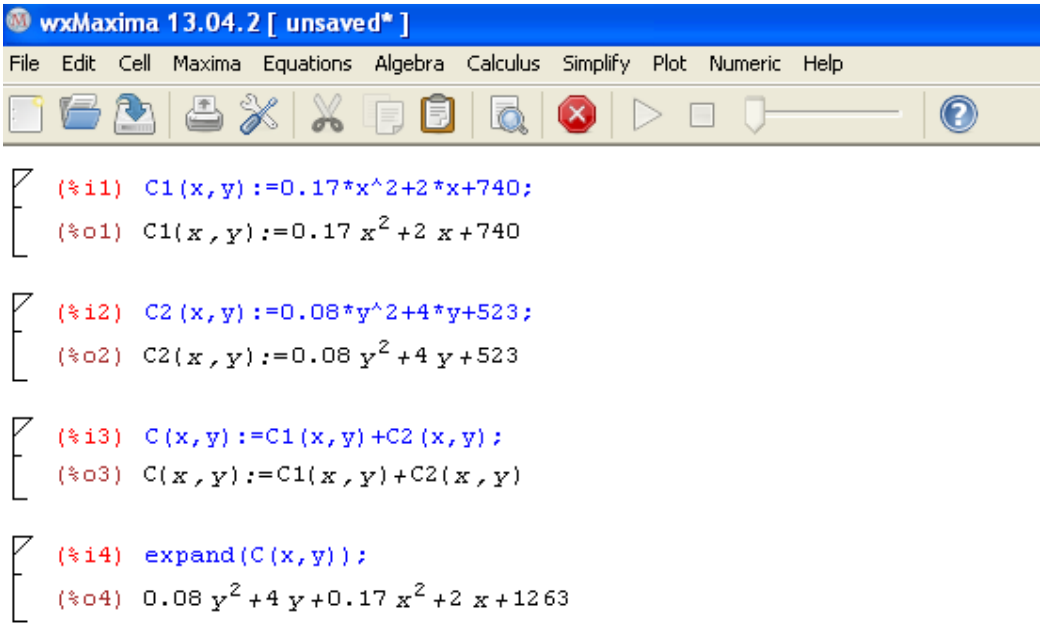
```
expand(C(x,y));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$0.08y^2+4y+0.17x^2+2x+1263$$

(vidjeti Sliku 4.9.).

Slika 4.9. Zadavanje funkcija C_1, C_2 i C u Primjeru 2.



Dakle,

$$C(x, y) = 0.08 \cdot y^2 + 4 \cdot y + 0.17 \cdot x^2 + 2 \cdot x + 1263.$$

Neka je $R(x,y)$ funkcija koja označava prihode od prodaje obiju vrsta vina. Jedinična cijena vina „*Blatina*“ je 28 n.j., pa se prodajom x litara toga vina ostvari prihod od $28 \cdot x$ n.j. Jedinična cijena vina „*Crljenak*“ je 27 n.j., pa se prodajom y litara toga vina ostvari prihod od $27 \cdot y$ n.j. Stoga je ukupan prihod nastao prodajom obiju vrsta vina:

$$R(x, y) = 28 \cdot x + 27 \cdot y.$$

Neka je $P(x, y)$ funkcija ukupne dobiti nastale prodajom obiju vrsta vina. Tada je:

$$P(x, y) = R(x, y) - C(x, y).$$

U novi redak programa *Maxima* utipka se:

```
R(x,y):=28*x+27*y;
```

Pritisne se *Enter*. U novi redak se utipka:

```
P(x,y):=R(x,y)-C(x,y);
```

Pritisne se *Enter*. Za ispis propisa funkcije P u novi redak se utipka:

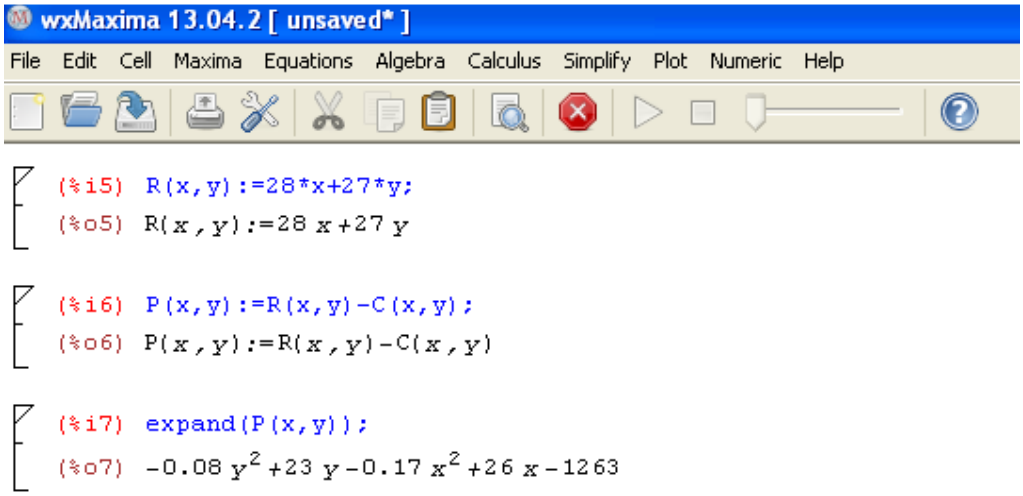
```
expand(P(x,y));
```

Pritisne se *Enter*. *Maxima* će ispisati:

$$-0.08y^2+23y-0.17x^2+26x-1263$$

(vidjeti Sliku 4.10.).

Slika 4.10. Zadavanje funkcija R i P u Primjeru 2.



Dakle,

$$P(x, y) = -0.17 \cdot x^2 - 0.08 \cdot y^2 + 26 \cdot x + 23 \cdot y - 1263.$$

Da bi se odredili lokalni ekstremi funkcije P, najprije je potrebno odrediti obje parcijalne derivacije te funkcije. Neka su P_x i P_y redom parcijalna derivacija funkcije P po varijabli x, odnosno varijabli y.

Najprije se određuje P_x. U novi redak programa *Maxima* utipka se:

```
Px(x,y):=diff(P(x,y),x);
```

Pritisne se tipka *Enter*. Za ispis propisa funkcije P_x u novi redak se utipka:

```
expand(Px(x,y));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
26-0.34*x
```

Dakle,

$$P_x(x, y) = -0.34 \cdot x + 26.$$

U nastavku se određuje P_y. U novi redak programa *Maxima* utipka se:

```
Py(x,y):=diff(P(x,y),y);
```

Pritisne se tipka *Enter*. Za ispis propisa funkcije P_y u novi redak se utipka:

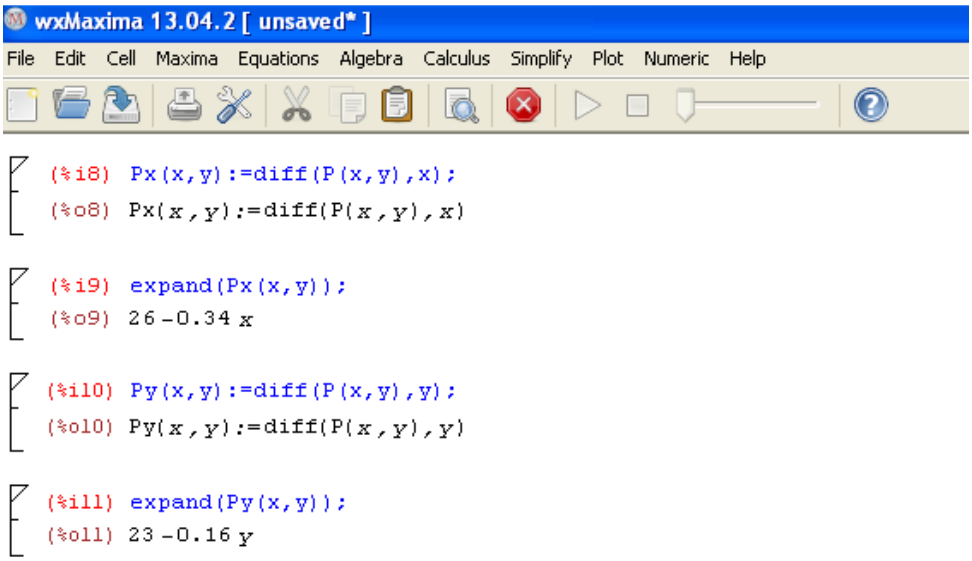
```
expand(Py(x,y))
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
23-0.16y
```

(vidjeti Sliku 4.11.).

Slika 4.11. Određivanje funkcija P_x i P_y u Primjeru 2.



Dakle,

$$P_y(x, y) = -0.16 \cdot y + 23.$$

Stacionarne točke su rješenja sustava jednadžbi:

$$P_x(x, y) = 0$$

$$P_y(x, y) = 0.$$

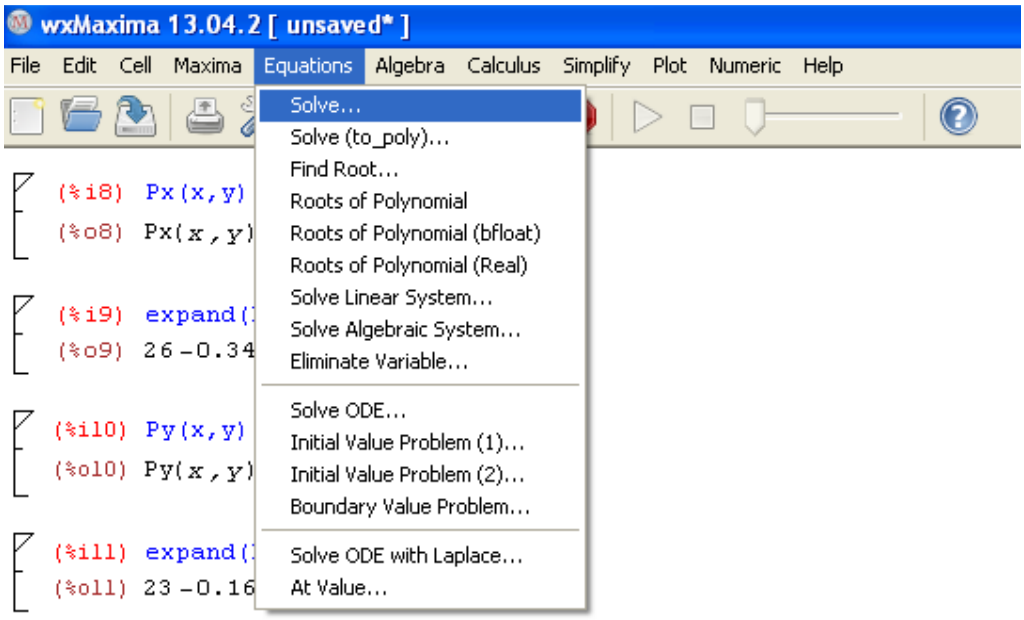
U ovom slučaju to je sustav:

$$-0.34 \cdot x + 26 = 0$$

$$-0.16 \cdot y + 23 = 0.$$

U izborniku se odabere opcija *Equations* i njezina podopcija *Solve* (vidjeti Sliku 4.12.).

Slika 4.12. Opcija *Equations* i njezina podopcija *Solve*



U pravokutnik pored natpisa *Equation*(*s*) u prozorčiću *Solve* upisuje se:

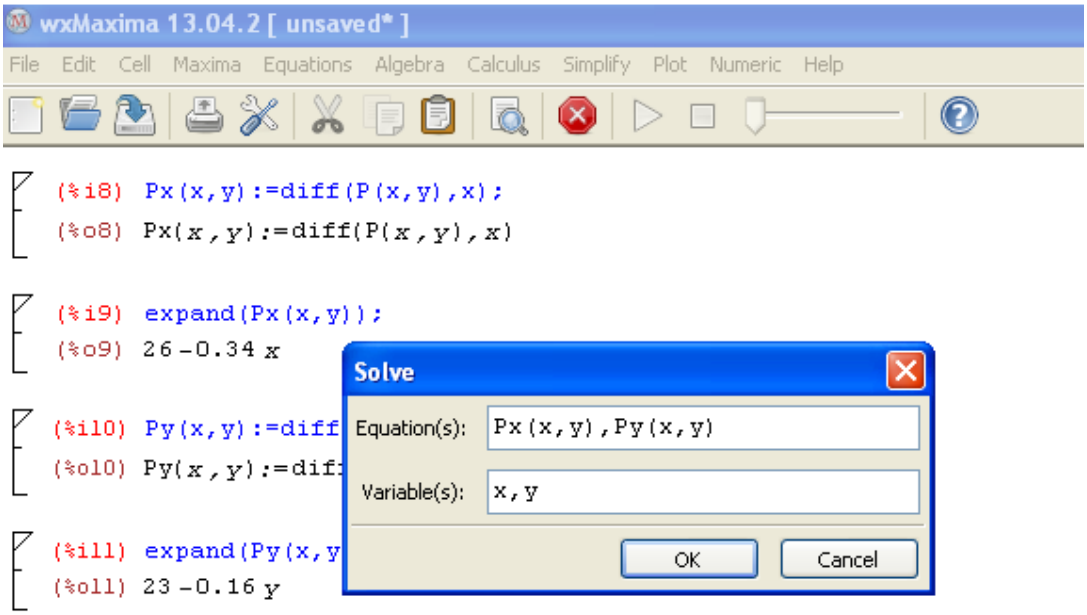
```
Px(x,y),Py(x,y)
```

U pravokutnik pored natpisa *Variable*(*s*) u prozorčiću *Solve* upisuje se:

```
x,y
```

(vidjeti Sliku 4.13.).

Slika 4.13. Upisivanje u podopciju *Solve* u Primjeru 2.



Klikne se na OK. *Maxima* će ispisati:

```
rat: replaced -0.34 by -17/50 = -0.34
rat: replaced -0.16 by -4/25 = -0.16
[[x=1300/17,y=575/4]]
```

Dakle, stacionarna točka je $T = \left(\frac{1300}{17}, \frac{575}{4}\right)$.

Treba provjeriti je li točka *T* ujedno i točka lokalnoga maksimuma funkcije *P*. U tu se svrhu računa pripadna *Hesseova* matrica. U novi redak programa *Maxima* utipka se:

```
H:hessian(P(x,y),[x,y]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$\begin{bmatrix} -0.34 & 0 \\ 0 & -0.16 \end{bmatrix}.$$

Dakle,

$$H = \begin{bmatrix} -0.34 & 0 \\ 0 & -0.16 \end{bmatrix}$$

Potom se odredi determinanta matrice *H*. U novi redak programa *Maxima* utipka se:

```
determinant(H);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
0.0544
```

Zaključuje se da je H_{11} = -0.34 < 0 i det(H) = 0.0544 > 0, pa je T točka lokalnoga maksimuma funkcije P.

Koordinate točke T treba zapisati kao decimalne brojeve. U novi redak programa *Maxima* utipka se:

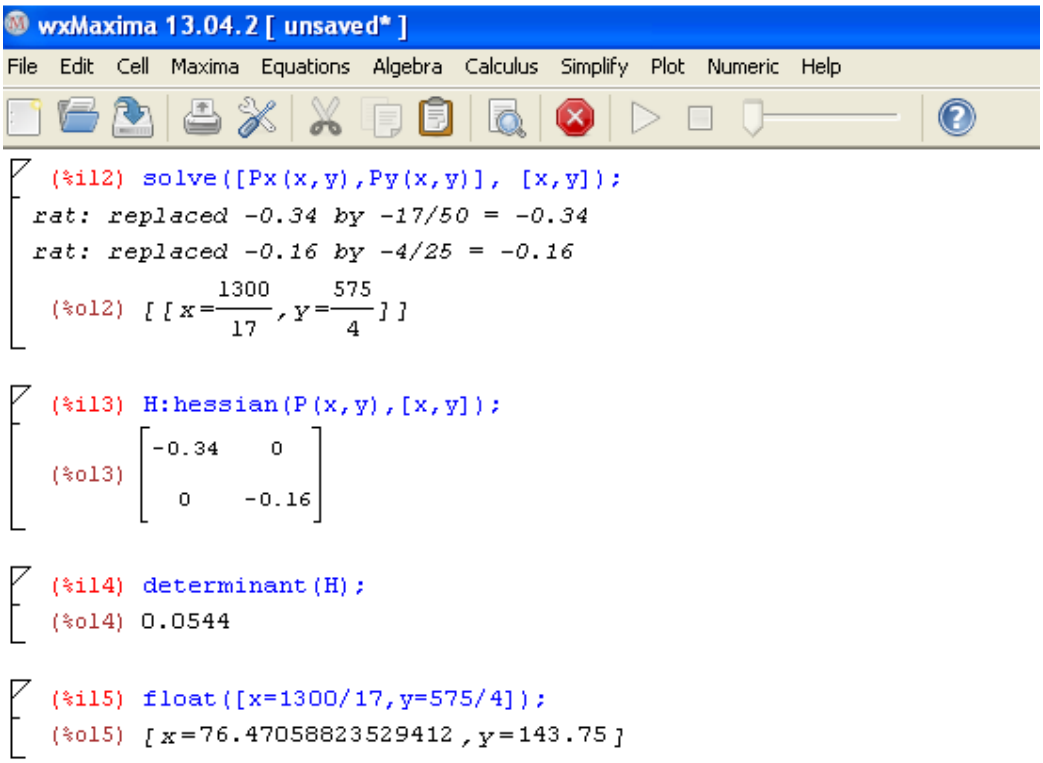
```
float([x=1300/17,y=575/4]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
[x=76.47058823529412,y=143.75]
```

(vidjeti Sliku 4.14.).

Slika 4.14. Izračun stacionarne točke i pripadne Hesseove matrice u Primjeru 2.



Budući da optimalne vrijednosti trebaju biti nenegativni cijeli brojevi, mogući su sljedeći slučajevi:

1.) $x^* = 76$, $y^* = 143$;

2.) $x^* = 76$, $y^* = 144$;

3.) $x^* = 77$, $y^* = 143$;

4.) $x^* = 77$, $y^* = 144$.

Optimalna dobit jednaka je najvećoj od četiriju vrijednosti: $P(76, 143)$, $P(76, 144)$, $P(77, 143)$ i $P(77, 144)$. U novi redak programa *Maxima* utipka se:

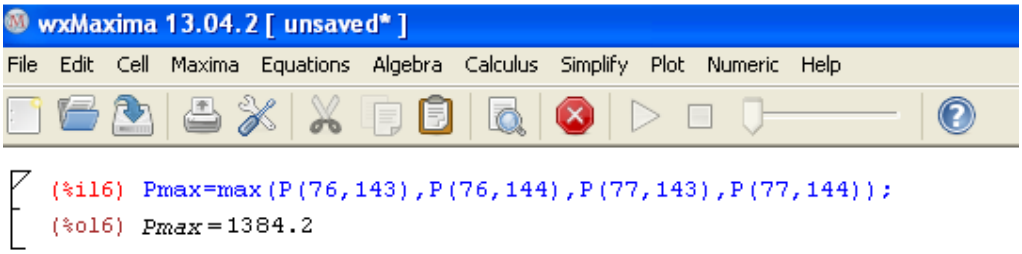
```
Pmax=max(P(76,143),P(76,144),P(77,143),P(77,144));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
Pmax=1384.2
```

(vidjeti Sliku 4.15.).

Slika 4.15. Izračun optimalne dobiti u Primjeru 2.



Dakle, maksimalna dobit iznosi 1384.2 n.j. Lako se provjeri da je $P(77, 144) = 1384.2$, pa slijedi da treba proizvesti 77 litara vina „*Blatina*“ i 144 litre vina „*Crljenak*“.

4.3. Primjer 3.

Tvrtka „*Chipi Chips d.d.*“ iz Zagreba proizvodi čips na dvije lokacije: Sesvete i Dugo Selo. U Sesvetama se proizvodi čips „*Chipy*“ čija je jedinična cijena 148.6 n.j., a troškovi proizvodnje $0.001998 \cdot x^2 + 3.813 \cdot x + 531.6$, gdje je x broj pakiranja čipsa (iskazan u komadima). U Dugom Selu se proizvodi čips „*Chipsy*“ čija je jedinična cijena također 148.6 n.j., a troškovi proizvodnje $0.005698 \cdot y^2 + 4.045 \cdot y + 349.6$, gdje je y broj pakiranja čipsa (iskazan u komadima). Treba odrediti količine čipsa koje treba proizvesti na svakoj lokaciji tako da ukupna dobit od prodaje obiju vrsta čipsa bude maksimalna uz uvjet da se u Dugom Selu može proizvesti najviše 11 000 komada čipsa. Pritom optimalne vrijednosti varijabli x i y moraju biti nenegativni cijeli brojevi. [11]

Ukupna dobit dobiva se oduzimanjem ukupnih troškova proizvodnje obiju vrsta čipsa od prihoda nastaloga prodajom obiju vrsta čipsa. Ukupni troškovi proizvodnje obiju vrsta čipsa jednaki su zbroju troškova proizvodnje čipsa „*Chipy*“ i troškova proizvodnje čipsa „*Chipsy*“.

Neka je $C_1(x, y)$ funkcija koja označava troškove proizvodnje čipsa „*Chipy*“. Očito je:

$$C_1(x, y) = 0.001998 \cdot x^2 + 3.813 \cdot x + 531.6.$$

Neka je $C_2(x, y)$ funkcija koja označava troškove proizvodnje čipsa „*Chipsy*“. Očito je:

$$C_2(x, y) = 0.005698 \cdot y^2 + 4.045 \cdot y + 349.6$$

pa ukupni troškovi proizvodnje obiju vrsta čipsa iznose:

$$C(x, y) = C_1(x,y) + C_2(x, y).$$

Ove funkcije upisuju se u program *Maxima*. Nakon pokretanja programa u prvi redak se upisuje funkcija koja označava troškove proizvodnje čipsa „*Chipy*“ :

```
C1(x,y):=0.001998*x^2+3.813*x+531.6;
```

Pritisne se tipka *Enter*. Potom se u novi redak upisuje funkcija koja označava troškove proizvodnje čipsa „*Chipsy*“:

```
C2(x,y):=0.005698*y^2+4.045*y+349.6;
```

Pritisne se tipka *Enter*. U novi redak upisuje se funkcija ukupnih troškova proizvodnje obiju vrsta čipsa:

```
C(x,y):=C1(x,y)+C2(x,y);
```

Za ispis propisa funkcije $C(x, y)$ u novi redak se utipka:

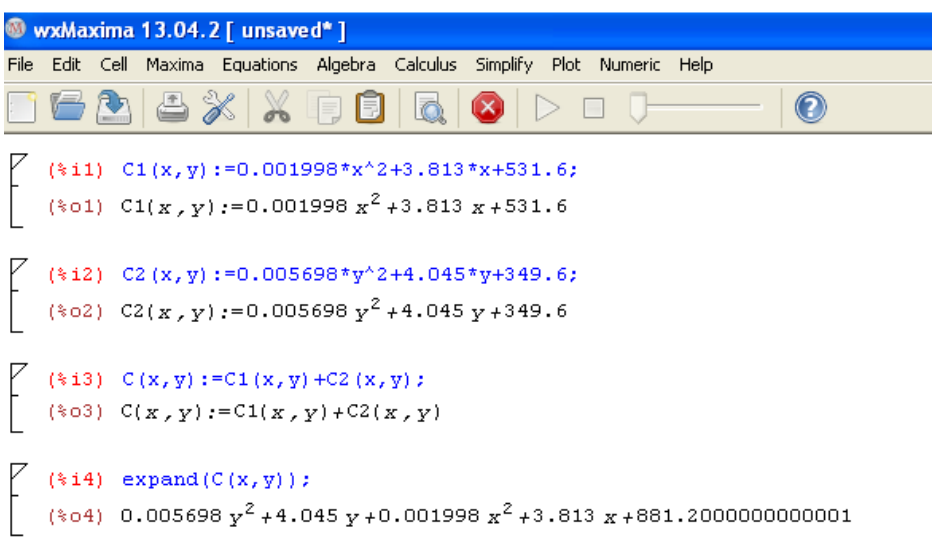
```
expand(C(x,y));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$0.005698y^2+4.045y+0.001998x^2+3.813x+881.2000000000001$$

(vidjeti Sliku 4.16.).

Slika 4.16. Zadavanje funkcija C_1, C_2 i C u Primjeru 3.



Dakle,

$$C(x, y) = 0.005698 \cdot y^2 + 4.045 \cdot y + 0.001998 \cdot x^2 + 3.813 \cdot x + 881.2.$$

Analitički se lako dobije da je slobodni član u gornjem polinomu jednak 881.2, dok *Maxima* ispisuje 881.2000000000001. Pogreška izračuna je reda veličine 10^{-13} i posljedica je načina pohrane simboličkih objekata (u ovom je slučaju to polinom C) u memoriji programa.

Neka je $R(x,y)$ funkcija koja označava prihode od prodaje obiju vrsta čipsa. Jedinična cijena čipsa „*Chipy*“ je 148.6 n.j., pa se prodajom x komada toga čipsa ostvari prihod od $148.6 \cdot x$ n.j. Jedinična cijena čipsa „*Chipsy*“ je također 148.6 n.j., pa se prodajom y komada toga čipsa ostvari prihod od $148.6 \cdot y$ n.j. Stoga je ukupan prihod nastao prodajom obiju vrsta čipsa:

$$R(x, y) = 148.6 \cdot x + 148.6 \cdot y.$$

Neka je $P(x, y)$ funkcija ukupne dobiti nastale prodajom obiju vrsta čipsa. Tada je:

$$P(x, y) = R(x, y) - C(x, y).$$

U novi redak programa *Maxima* utipka se:

```
R(x,y):=148.6*x+148.6*y;
```

Pritisne se *Enter*. U novi redak se utipka:

```
P(x,y):=R(x,y)-C(x,y);
```

Pritisne se *Enter*. Za ispis propis funkcije P u novi redak se utipka:

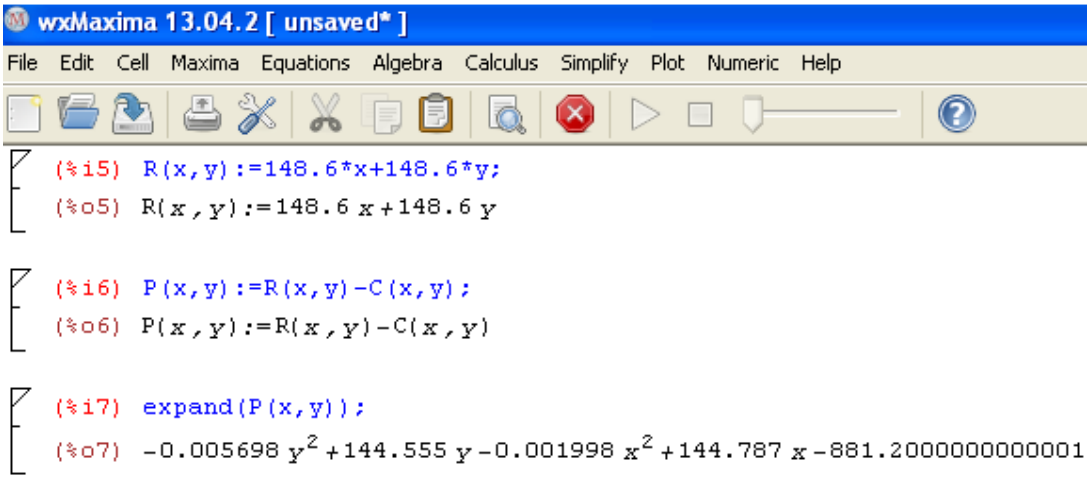
```
expand(P(x,y));
```

Pritisne se *Enter*. *Maxima* će ispisati:

$$-0.005698y^2+144.555y-0.001998x^2+144.787x-881.2000000000001$$

(vidjeti Sliku 4.17.).

Slika 4.17. Zadavanje funkcija *R* i *P* u Primjeru 3.



Dakle,

$P(x, y) = -0.001998 \cdot x^2 - 0.005698 \cdot y^2 + 144.787 \cdot x + 144.555 \cdot y - 881.2.$

Da bi se odredili lokalni ekstremi funkcije *P*, najprije je potrebno odrediti obje parcijalne derivacije te funkcije. Neka su P_x i P_y redom parcijalna derivacija funkcije *P* po varijabli *x*, odnosno varijabli *y*.

Najprije se određuje P_x. U novi redak programa *Maxima* utipka se:

```
Px(x,y):=diff(P(x,y),x);
```

Pritisne se tipka *Enter*. Za ispis propisa funkcije P_x u novi redak se utipka:

```
expand(Px(x,y));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
144.787-0.003996*x
```

Dakle,

$$P_x(x, y) = -0.003996 \cdot x + 144.787.$$

U nastavku se određuje P_y. U novi redak programa *Maxima* utipka se:

```
Py(x,y):=diff(P(x,y),y);
```

Pritisne se tipka *Enter*. Za ispis propisa funkcije P_y u novi redak se utipka:

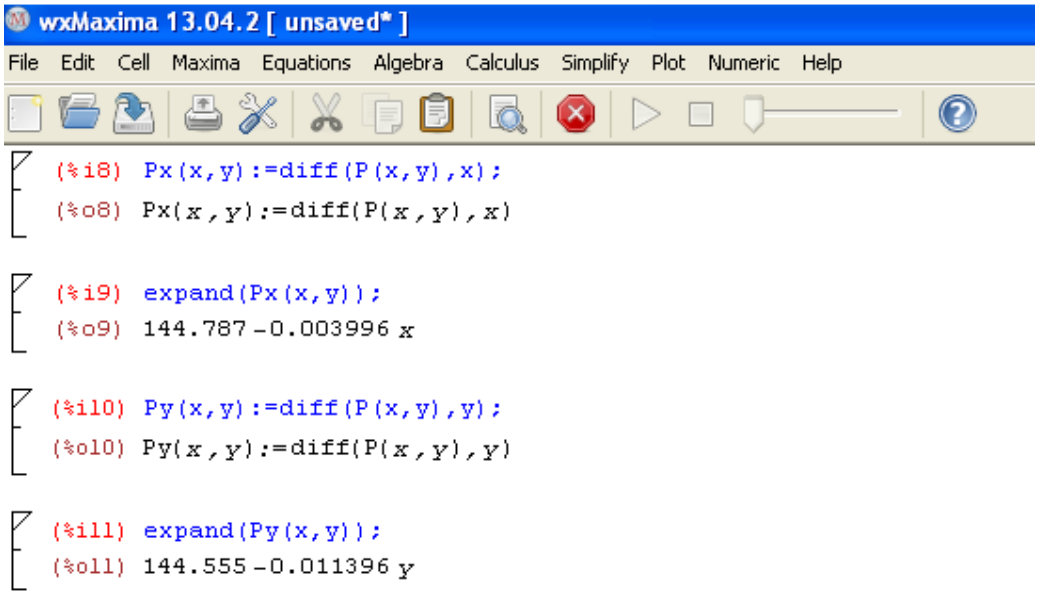
```
expand(Py(x,y))
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
144.555-0.011396y
```

(vidjeti Sliku 4.18.).

Slika 4.18. Određivanje funkcija P_x i P_y u Primjeru 3.



Dakle,

$$P_y(x, y) = -\,0.011396 \cdot y + 144.555.$$

Stacionarne točke su rješenja sustava jednadžbi:

$$P_x(x, y) = 0$$

$$P_y(x, y) = 0.$$

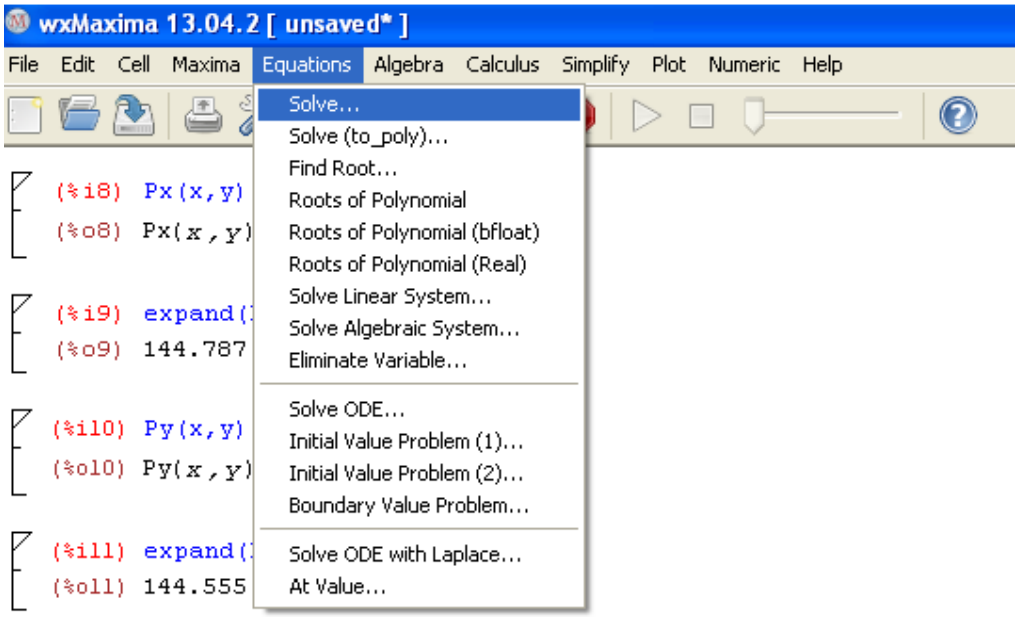
U ovom slučaju to je sustav:

$$-\,0.003996 \cdot x + 144.787 = 0$$

$$-\,0.003996 \cdot x + 144.787 = 0.$$

U izborniku se odabere opcija *Equations* i njezina podopcija *Solve* (vidjeti Sliku 4.19.).

Slika 4.19. Opcija *Equations* i njezina podopcija *Solve*



U pravokutnik pored natpisa *Equation*(*s*) u prozorčiću *Solve* upisuje se:

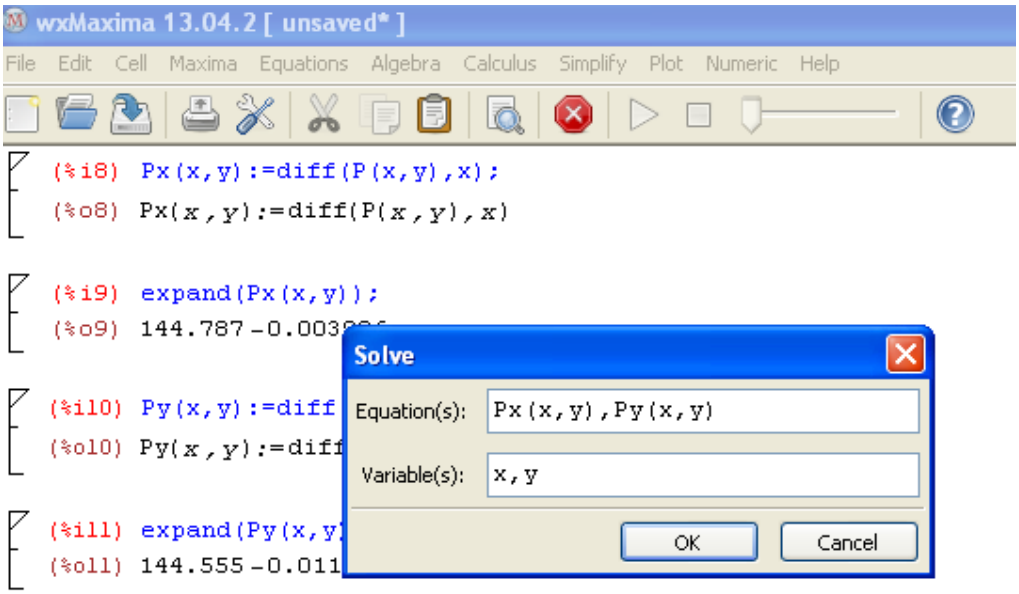
```
Px(x,y),Py(x,y)
```

U pravokutnik pored natpisa *Variable*(*s*) u prozorčiću *Solve* upisuje se:

```
x,y
```

(vidjeti Sliku 4.20.).

Slika 4.20. Upisivanje u podopciju *Solve* u Primjeru 3.



Klikne se na OK. *Maxima* će ispisati:

```
rat: replaced 144.787 by 144787/1000 = 144.787
rat: replaced -0.003996 by -999/250000 = -0.003996
rat: replaced 144.555 by 28911/200 = 144.555
rat: replaced -0.011396 by -2849/250000 = -0.011396
[[x=36196750/999,y=36138750/2849]]
```

Dakle, stacionarna točka je $T = \left(\frac{36196750}{99}, \frac{36138750}{2849} \right)$.

Treba provjeriti je li točka *T* ujedno i točka lokalnoga maksimuma funkcije *P*. U tu se svrhu računa pripadna *Hesseova* matrica. U novi redak programa *Maxima* utipka se:

```
H:hessian(P(x,y),[x,y]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$\begin{bmatrix} -0.003996 & 0 \\ 0 & -0.011396 \end{bmatrix}.$$

Dakle,

$$H = \begin{bmatrix} -0.003996 & 0 \\ 0 & -0.0011396 \end{bmatrix}.$$

Potom se odredi determinanta matrice *H*. U novi redak programa *Maxima* utipka se:

```
determinant(H);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$4.5538416000000001\ 10^{-5}$$

Zaključuje se da je H_{11} = -0.003996 < 0 i det(*H*) = 0.000045538416000000001 > 0, pa je *T* točka lokalnoga maksimuma funkcije *P*.

S manjom preciznošću izračuna (npr. s točnošću od 10^{-2} ili 10^{-3}) bi se mogao izvesti zaključak da je det(*H*) = 0 i da *T* nije lokalni ekstrem funkcije *P*. Zbog toga ovaj primjer izvrsno ukazuje na nužnost računanja „kontrolnih“ podataka (konkretno, determinante matrice *H*) na veći broj decimalnih mjesta.

Koordinate točke *T* treba zapisati kao decimalne brojeve. U novi redak programa *Maxima* utipka se:

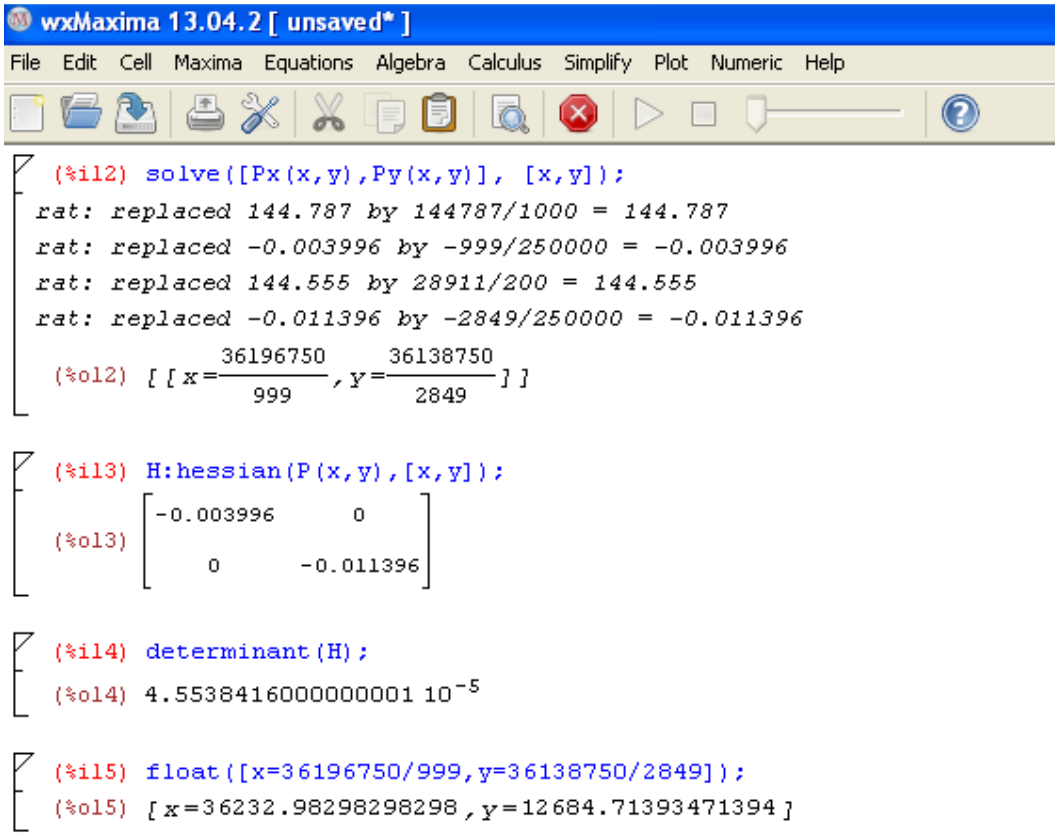
```
float([x=36196750/999,y=36138750/2849]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
[x=36232.98298298298,y=12684.71393471394]
```

(vidjeti Sliku 4.21.).

Slika 4.21. Izračun stacionarne točke i pripadne Hesseove matrice u Primjeru 3.



Budući da se na drugoj lokaciji može proizvesti najviše 11 000 komada čipsa i da optimalne vrijednosti trebaju biti nenegativni cijeli brojevi, mogući su sljedeći slučajevi:

1.) $x^* = 36232$, $y^* = 11000$;

2.) $x^* = 36233$, $y^* = 11000$.

Optimalna dobit jednaka je najvećoj od dviju vrijednosti: $P(36232, 11000)$, $P(36233, 11000)$. U novi redak programa *Maxima* utipka se:

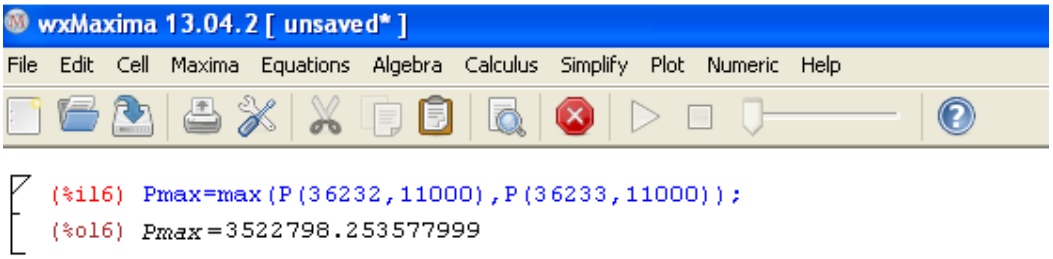
```
Pmax=max(P(36232,11000),P(36233,11000));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
Pmax=3522798.253577999
```

(vidjeti Sliku 4.22.).

Slika 4.22. Izračun optimalne dobiti u Primjeru 3.



Dakle, maksimalna dobit iznosi 3 522 798.25 n.j. Lako se provjeri da je $P(36233, 11000) = 3\ 522\ 798.25$, pa slijedi da treba proizvesti 36233 komada čipsa „*Chipy*“ i 11000 komada čipsa „*Chipsy*“.

Iako su prodajne cijene obiju vrsta čipsa jednake, optimalan plan predviđa više nego trostruku proizvodnju čipsa „*Chipy*“ u odnosu na proizvodnju čipsa „*Chipsy*“. Razlog tome su funkcije troškova C_1 i C_2. Proizlazi da bi, radi uravnoteženja proizvodnje na objema lokacijama, tvrtka trebala razmotriti reduciranje troškova proizvodnje u Dugom Selu. To se danas uistinu i čini u praksi, pri čemu se najprije nastoje reducirati ukupni troškovi radne snage, odnosno plaće radnika.

4.4. Primjer 4.

Tvrtka „*Štilko d.d.*“ iz Požege bavi se proizvodnjom motornih pila. Uprava tvrtke namjerava izgraditi novo skladište tako da zbroj udaljenosti od toga skladišta do triju najvažnijih klijenata tvrtke bude što manji. Ako se pretpostavi da se klijenti nalaze u točkama $A = (2,10)$, $B = (0,0)$ i $C = (16,1)$, treba odrediti u kojoj točki W treba izgraditi skladište. [12]

Neka je $W = (x,y)$ tražena točka. Primijeni se formula za izračunavanje udaljenosti između dviju točaka $T_1 = (x_1, y_1)$ i $T_2 = (x_2, y_2)$ u pravokutnom koordinatnom sustavu u ravnini:

$$d(T_1,T_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}\ .$$

U zadatku se traži minimiziranje zbroja $d(A, W) + d(B, W) + d(C, W)$. Taj zbroj će poprimiti najmanju vrijednost ako i samo ako zbroj $[d(A, W)]^2 + [d(B, W)]^2 + [d(C, W)]^2$ bude poprimio najmanju vrijednost.

Kvadrat udaljenosti između točaka $A = (2, 10)$ i $W = (x,y)$ jednak je:

$$[d(A, W)]^2 = (x - 2)^2 + (y - 10)^2.$$

Kvadrat udaljenosti između točaka $B = (0, 0)$ i $W(x,y)$ jednak je:

$$[d(B, W)]^2 = x^2 + y^2.$$

Kvadrat udaljenosti između točaka $C = (16, 1)$ i $W(x,y)$ jednak je:

$$[d(C, W)]^2 = (x - 16)^2 + (y - 1)^2.$$

Neka je $S(x, y)$ funkcija koja označava zbroj gornjih triju kvadrata udaljenosti. Tada je:

$$S(x,y) = (x - 2)^2 + (y - 10)^2 + (x^2 + y^2) + (x - 16)^2 + (y - 1)^2.$$

U program *Maxima* unosi se:

```
S(x,y):=(x-2)^2+(y-10)^2+(x^2+y^2)+(x-16)^2+(y-1)^2;
```

Pritisne se *Enter*. Za ispis propisa funkcije S u novi redak se utipka:

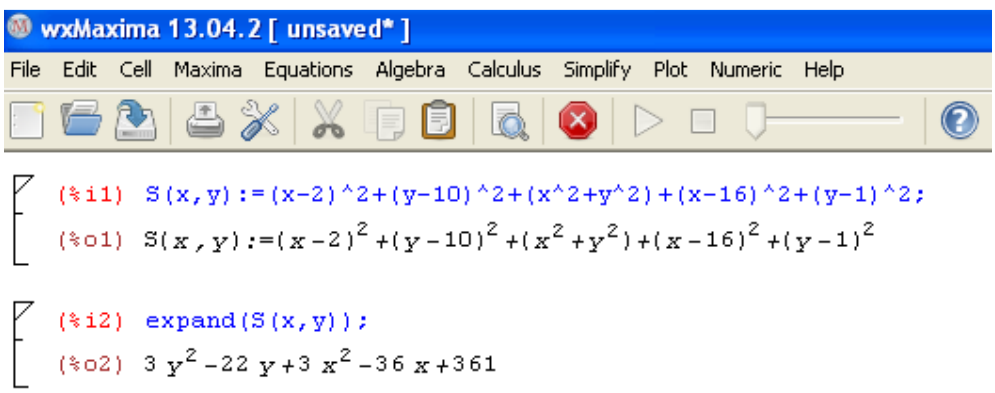
```
expand(S(x,y));
```

Pritisne se *Enter*. *Maxima* će ispisati:

$$3y^2-22y+3x^2-36x+361$$

(vidjeti Sliku 4.23.).

Slika 4.23. Zadavanje funkcije S u Primjeru 4.



Dakle,

$$S(x, y) = 3 \cdot x^2 - 22 \cdot y^2 - 36 \cdot x - 22 \cdot y + 361.$$

Da bi se odredili lokalni ekstremi funkcije S, najprije je potrebno odrediti obje parcijalne derivacije te funkcije. Neka su S_x i S_y redom parcijalna derivacija funkcije S po varijabli x, odnosno varijabli y.

Najprije se određuje S_x. U novi redak programa *Maxima* utipka se:

```
Sx(x,y):=diff(S(x,y),x);
```

Pritisne se tipka *Enter*. Za ispis propisa funkcije S_x u novi redak se utipka:

```
expand(Sx(x,y));
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
6x-36
```

Dakle,

$$S_x(x, y) = 6 \cdot x - 36.$$

U nastavku se određuje S_y. U novi redak programa *Maxima* utipka se:

```
Sy(x,y):=diff(S(x,y),y);
```

Pritisne se tipka *Enter*. Za ispis propisa funkcije S_y u novi redak se utipka:

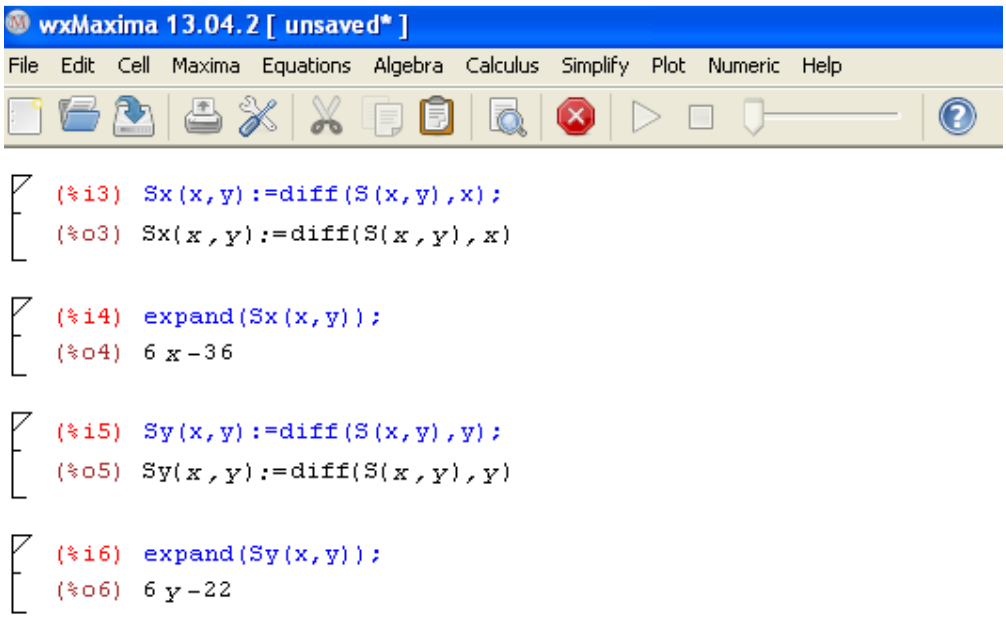
```
expand(Sy(x,y))
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
6y-22
```

(vidjeti Sliku 4.24.).

Slika 4.24. Određivanje funkcija S_x i S_y u Primjeru 4.



Dakle,

$$S_y(x, y) = 6 \cdot y - 22.$$

Stacionarne točke su rješenja sustava jednadžbi:

$$S_x(x, y) = 0$$
$$S_y(x, y) = 0.$$

U ovom slučaju to je sustav:

$$6 \cdot x - 36 = 0$$
$$6 \cdot y - 22 = 0$$

Njegova rješenja dobijemo izravno (bez korištenja programa *Maxima*):

$$x = 6$$
$$y = \frac{3}{11}.$$

Dakle, stacionarna točka je $T = \left(6, \frac{3}{11}\right)$.

Treba provjeriti je li točka *T* točka lokalnoga maksimuma ili minimuma funkcije *P*. U tu se svrhu računa pripadna *Hesseova* matrica. U novi redak programa *Maxima* utipka se:

```
H:hessian(S(x,y),[x,y]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$\begin{bmatrix} 6 & 0 \\ 0 & 6 \end{bmatrix}.$$

Dakle,

$$H = \begin{bmatrix} 6 & 0 \\ 0 & 6 \end{bmatrix}.$$

Potom se odredi determinanta matrice *H*. U novi redak programa *Maxima* utipka se:

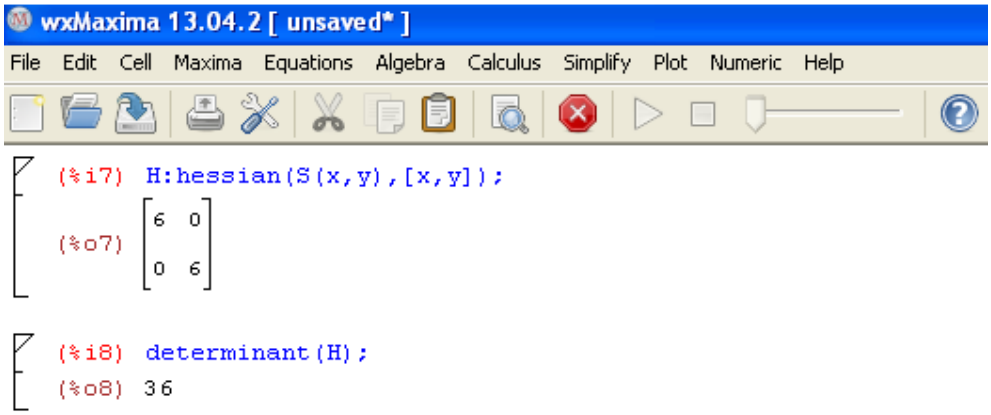
```
determinant(H);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
36
```

Zaključuje se da je $H_{11} = 6 > 0$ i $\det(H) = 36 > 0$, pa je *T* točka lokalnoga minimuma funkcije *S* (vidjeti Sliku 4.25.).

Slika 4.25. Izračun Hesseove matrice za stacionarnu točku iz Primjera 4.



U cilju smanjivanja udaljenosti od *W* do *A*, *B* i *C* skladište tvrtke „*Štilko d.d.*“ treba izgraditi u točki $W=\left(6,\ \frac{3}{11}\right)$.

Budući da su poznate koordinate točke *W*, treba izračunati zbroj udaljenosti točke *W* od točaka *A*, *B* i *C* koristeći formulu $d(T_1,T_2)=\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$. Treba zasebno izračunati *d*(*A*, *W*), *d*(*B*, *W*) i *d*(*C*, *W*), pa zbrojiti dobivene vrijednosti. Koristeći navedenu formulu dobiva se:

$$d(A,W)+d(B,W)+d(C,W)=\sqrt{(6-2)^2+\left(\frac{3}{11}-10\right)^2}+\sqrt{6^2+\left(\frac{3}{11}\right)^2}+\sqrt{(6-16)^2+\left(\frac{3}{11}-1\right)^2}\ .$$

U novi redak programa *Maxima* utipka se:

```
d(x1,y1,x2,y2):=(x2-x1)^2+(y2-y1)^2;
```

Pritisne se tipka *Enter*. Potom se utipka:

```
d=d(2,10,6,3/11)+d(0,0,6,3/11)+d(16,1,6,3/11);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$d=\frac{\sqrt{13385}}{11}+\frac{2\sqrt{3041}}{11}+\frac{3\sqrt{485}}{11}.$$

Da bi se dobila približna vrijednost broja *d*, u novi redak se utipka:

```
float(d=sqrt(13385)/11+(2*sqrt(3041))/11+(3*sqrt(485))/11);
```

Pritom se argument funkcije `float` prekopira iz neposredno prethodnoga retka koristeći uobičajenu *Copy* – *Paste* proceduru. Dobiva se:

```
d=26.55020298387228
```

Dakle, traženi optimalan zbroj svih udaljenosti iznosi približno 26.55 jedinica duljine.

Sve četiri točke *A*, *B*, *C* i *W*, prikazane su u pravokutnom koordinatnom sustavu u ravnini. Treba primijetiti da se optimalna točka *W* nalazi izvan trokuta *ABC* (vidjeti Sliku 4.26.).

Slika 4.26. Prikaz točaka *A*, *B*, *C* i *W* u pravokutnom koordinatnom sustavu u ravnini

4.5. Primjer 5.

Procjenjuje se da je tjedni output u tvornici čokolade „*Čokolicija d.d.*“ iz Velike jednak $Q(x, y) = 1.731 \cdot x + 925 \cdot y + x^2 \cdot y - 2.7 \cdot x^2 - 1.3 \cdot y^{\frac{3}{2}}$ jedinica, gdje je x broj kvalificiranih radnika, a y broj nekvalificiranih radnika zaposlenih u tvornici. Trenutna radna snaga se sastoji od 43 kvalificirana i 85 nekvalificiranih radnika. Treba izračunati tjedni output u svakom od sljedećih slučajeva:

a) u proizvodnji rade svi trenutno zaposleni radnici;

b) broj kvalificiranih radnika poveća se za 1, a broj nekvalificiranih radnika ostane nepromijenjen;

c) broj nekvalificiranih radnika poveća se za 1, a broj kvalificiranih radnika ostane nepromijenjen.

Potom treba izračunati relativnu promjenu tjednoga outputa u slučajevima **b)** i **c)** s obzirom na slučaj **a)**, te u sljedećim slučajevima:

d) broj kvalificiranih radnika se poveća za 1%, a broj nekvalificiranih radnika ostane nepromijenjen;

e) broj nekvalificiranih radnika se poveća za 1%, a broj kvalificiranih radnika ostane nepromijenjen. [13]

U programu *Maxima* najprije treba definirati funkciju *Q*. U prvi redak programa *Maxima* upiše se:

```
Q(x,y):=1.731*x+925*y+x^2*y-2.7*x^2-1.3*y^(3/2);
```

Pritisne se tipka *Enter*.

a)

Treba izračunati vrijednost *Q*(43, 85). U novi redak programa *Maxima* utipka se:

```
Q(43,85);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$230872.133-1.3\ 85^{3/2}$$

Dobiveni rezultat treba zapisati kao decimalni broj. U novi redak programa *Maxima* utipka se:

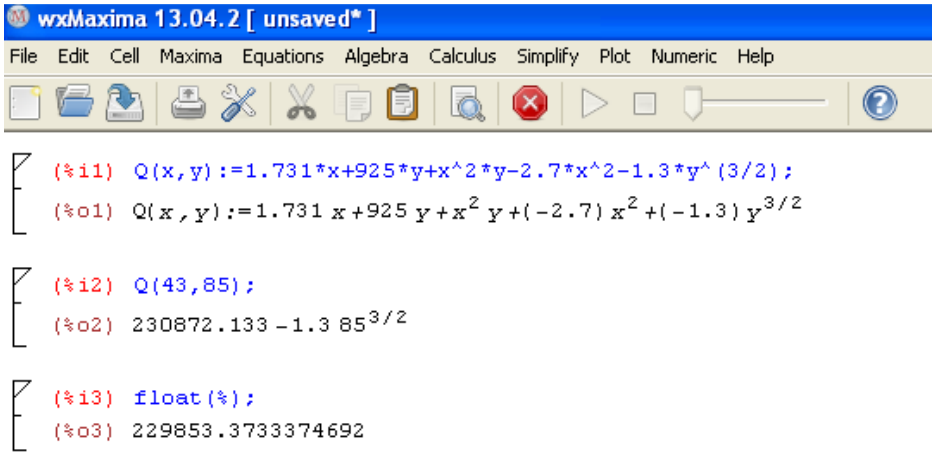
```
float(%);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
229853.3733374692
```

(vidjeti Sliku 4.27.)

Slika 4.27. Izračun vrijednosti *Q*(43, 85) u Primjeru 5.



Dakle, procijenjeni tjedni output u slučaju kad u proizvodnji rade svi trenutno zaposleni radnici iznosi približno 229 853 jedinice.

b)

Treba izračunati vrijednost $Q(44, 85)$. U novi redak programa *Maxima* utipka se:

```
Q(44,85);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$238033.964-1.3\ 85^{3/2}$$

Dobiveni rezultat treba zapisati kao decimalan broj. U novi redak se utipka:

```
float(%);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
237015.2043374691
```

Dakle, procijenjeni tjedni output u slučaju kad se broj kvalificiranih radnika poveća za 1 iznosi približno 237 015 jedinica.

Potrebno je i izračunati pripadnu relativnu promjenu u odnosu na **a)** zadatak. Ta je promjena definirana izrazom

$$r_1 = \frac{|Q(44,85) - Q(43,85)|}{Q(43,85)} \cdot 100.$$

U novi redak programa *Maxima* utipka se:

```
r1=(abs(Q(44,85)-Q(43,85))/Q(43,85))*100;
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$r1 = \frac{716183.0999999947}{230872.133-1.3\ 85^{3/2}}$$

Dobiveni rezultat treba zapisati kao decimalni broj. U novi redak programa *Maxima* utipka se:

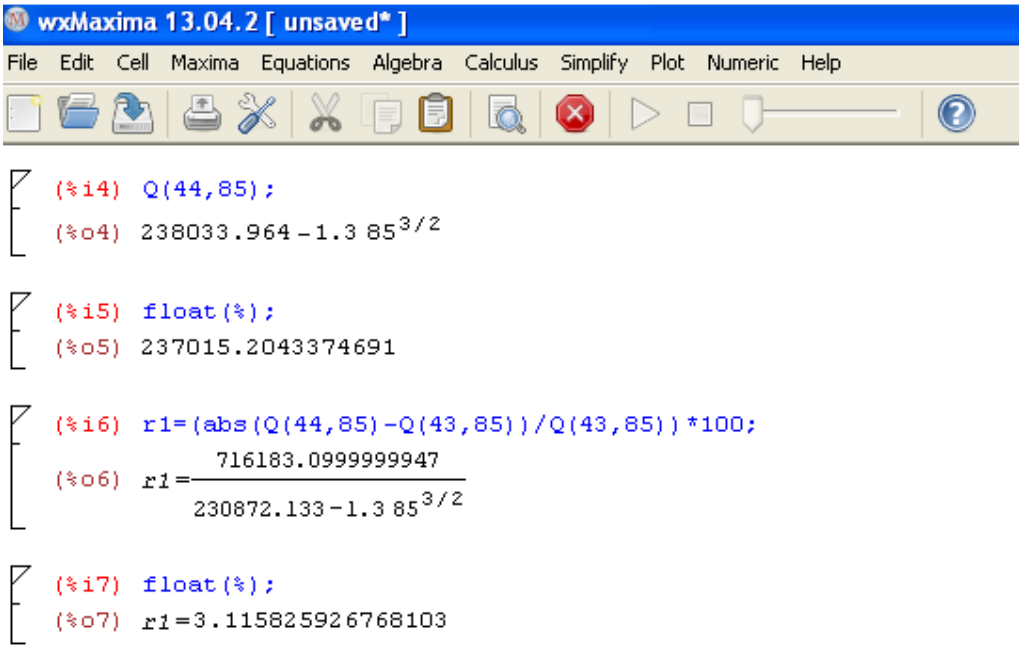
```
float(%);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
r1=3.115825926768103
```

(vidjeti Sliku 4.28.).

Slika 4.28. Izračun vrijednosti r_1 u Primjeru 5.



Interpretacija dobivenoga rezutlata je: ako se sadašnji broj kvalificiranih radnika poveća za 1, tjedni output će se povećati za približno 3.12 %.

c)

Treba izračunati $Q(43, 86)$. U novi redak programa *Maxima* utipka se:

```
Q(43,86);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$233646.133-1.3\ 86^{3/2}$$

Dobiveni rezultat treba zapisati kao „običan“ decimalan broj. U novi redak se utipka:

```
float(%);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
232609.3424522036
```

Dakle, procijenjeni tjedni output u slučaju kad se broj nekvalificiranih radnika poveća za 1 iznosi približno 232 609 jedinica.

Potrebno je izračunati pripadnu relativnu promjenu u odnosu na **a)** zadatak. Ta je promjena definirana izrazom:

$$r_2 = \frac{|Q(43,86) - Q(43,85)|}{Q(43,85)} \cdot 100 .$$

U novi redak programa *Maxima* utipka se:

```
r2=((Q(43,86)-Q(43,85))/Q(43,85))*100;
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$r2=\frac{100\left(-1.386^{3/2}+1.385^{3/2}+2774.0\right)}{230872.133-1.385^{3/2}}$$

Da bi se dobio decimalni zapis broja r_2, u novi redak se utipka:

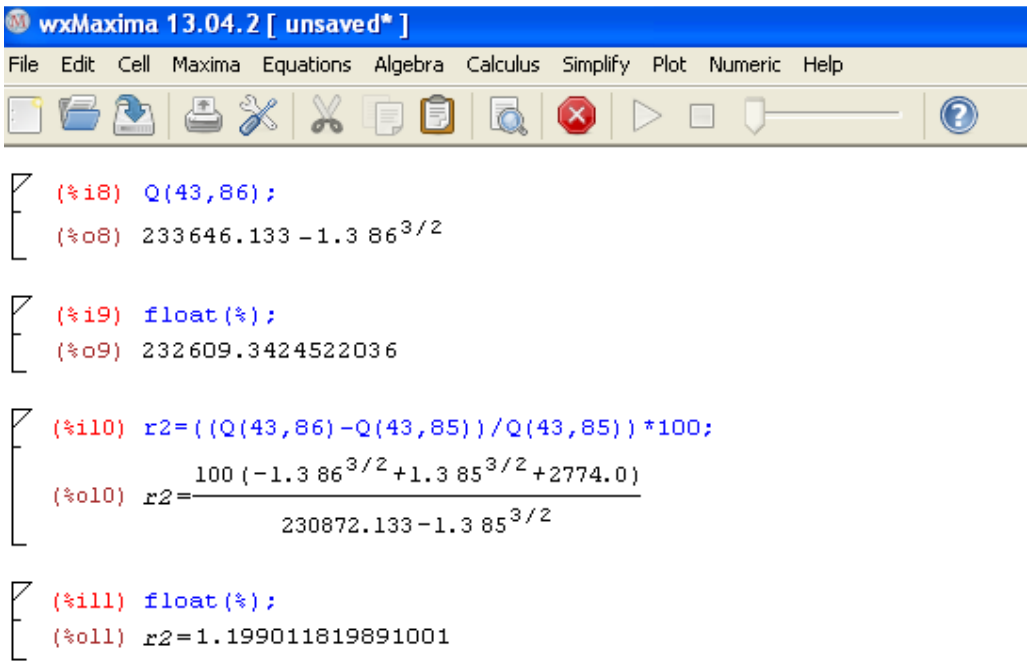
```
float(%)
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
r2=1.199011819891001
```

(vidjeti Sliku 4.29.)

Slika 4.29. Izračun vrijednosti r_2 u Primjeru 5.



Interpretacija dobivenoga rezultata je: ako se sadašnji broj nekvalificiranih radnika poveća za 1, tjedni output će se povećati za približno 1.2 %.

Relativne promjene u zadatcima **d)** i **e)** jednake su vrijednostima koeficijenata parcijalne elastičnosti $E_{Q,\ x}$ i $E_{Q,\ y}$ za x = 43 i y = 85. Ti su koeficijenti definirani izrazima:

$$E_1 := E_{Q,x}(43,85) = \frac{43}{Q(43,85)} \cdot Q_x(43,85)$$
$$E_2 := E_{Q,y}(43,85) = \frac{85}{Q(43,85)} \cdot Q_y(43,85) \quad .$$

Da bi se izračunali ti koeficijenti, potrebno je odrediti obje parcijalne derivacije funkcije *Q*. Neka su Q_x i Q_y redom parcijalna derivacija funkcije *Q* po varijabli *x*, odnosno varijabli *y*.

d)

Najprije se određuje Q_x. U novi redak programa *Maxima* utipka se:

```
diff(Q(x,y),x);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
2xy-5.4*x+1.731
```

Dakle,

$$Q_x(x, y) = 2 \cdot x \cdot y - 5.4 \cdot x + 1.731.$$

Potom se računa koeficijent E_1 prema gore navedenoj formuli. U novi redak programa *Maxima* utipka se:

```
E1=43/Q(43,85)*at(%,[x=43,y=85]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$E1 = \frac{304419.833}{230872.133 - 1.3\,85^{3/2}}$$

Izraz E_1 treba zapisati kao decimalni broj. U novi redak programa *Maxima* utipka se:

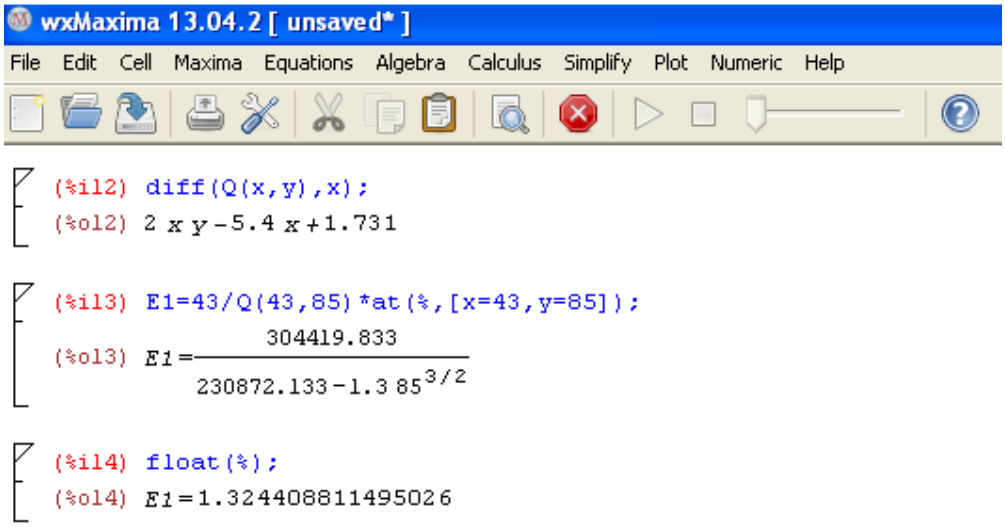
```
float(%);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
E1=1.324408811495026
```

(vidjeti Sliku 4.30.)

Slika 4.30. Izračun vrijednosti E_1 u Primjeru 5.



Interpretacija dobivenoga rezultata je: Ako se broj kvalificiranih radnika poveća za 1%, tjedni output će se očekivano povećati za približno 1.32 %.

e)

U nastavku se određuje Q_y. U novi redak programa *Maxima* utipka se:

```
diff(Q(x,y),y);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$-1.95\sqrt{y}+x^2+925$$

Dakle,

$$Q_y(x,y)=-1.95\cdot\sqrt{y}+x^2+925\,.$$

Potom se računa koeficijent E_2 prema gore navedenoj formuli. U novi redak programa *Maxima* utipka se:

```
E2=85/Q(43,85)*at(%,[x=43,y=85]);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

$$E2=\frac{85(2774-1.95\sqrt{85})}{230872.133-1.3\,85^{3/2}}$$

Izraz E_2 treba zapisati kao decimalni broj. U novi redak programa *Maxima* utipka se:

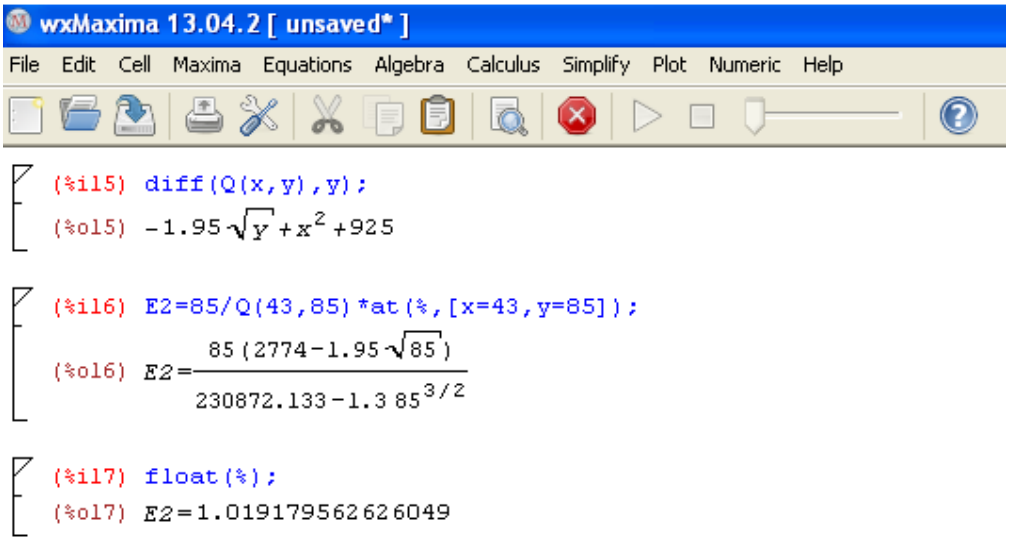
```
float(%);
```

Pritisne se tipka *Enter*. *Maxima* će ispisati:

```
E2=1.019179562626049
```

(vidjeti Sliku 4.31.)

Slika 4.31. Izračun vrijednosti E_2 u Primjeru 5.



Interpretacija dobivenoga rezultata je: ako se broj nekvalificiranih radnika poveća za 1%, tjedni output će se očekivano povećati za približno 1.02 %.

5. ZAKLJUČAK

U procesu poslovnoga odlučivanja vrlo često je potrebno izvršiti kvalitetnu analizu funkcija troškova, prihoda, profita i potražnje. U toj je analizi nužna primjena diferencijalnoga računa više varijabli. U tim se procesima vrlo često pojavljuje i problem određivanja lokalnih ekstrema neke od spomenutih funkcija. Iako se taj problem u nekim posebnim slučajevima može riješiti i analitički („klasično“), najčešće ga je nužno približno riješiti nekom od numeričkih metoda za rješavanje algebarskih jednadžbi ili sustava algebarskih jednadžbi. Da bi se izbjegli komplicirani izračuni koji se pojavljuju u tim metodama, korisno je primijeniti pogodne računalne programe koji će napraviti te izračune umjesto čovjeka – donositelja odluke.

U ovom je radu pokazana primjena računalnoga programa *Maxima* na rješavanje nekih tipičnih problema kvantitativne analize spomenutih ekonomskih funkcija i problema određivanja lokalnih ekstrema tih funkcija. Program *Maxima* je odabran jer na relativno jednostavan i brz način omogućuje približno izračunavanje lokalnih ekstrema realne funkcije dviju realnih varijabli.

Razvoj tehnologije i društva općenito uzrokuje sve veću potrebu za kvalitetnim i, za korisnike relativno jednostavnim računalnim programima koji će omogućiti kvalitetnu analizu tehnoloških procesa i donošenje optimalnih odluka. Treba istaknuti da nitko, pa ni najbolji računalni programi ne mogu zamijeniti čovjeka kao donositelja konačne odluke u procesu odlučivanja, ali mogu poslužiti kao vrlo korisni alati u svrhu smanjenja trajanja procesa odlučivanja i analize osjetljivosti donijetih odluka. Zbog toga je vrlo prikladno naučiti osnove rada s računalnim programima kao alatima u poslovnom odlučivanju, pri čemu nije potrebno znati kako program „tehnički“ funkcionira (tj. znati sadržaj kodova kojima su implementirane programske funkcije). Računalni program *Maxima* jedan je od takvih vrlo korisnih alata, pa se može očekivati njegova sve veća buduća primjena u procesima odlučivanja.

6. POPIS KRATICA I AKRONIMA

tj. – to jest

itd. – i tako dalje

n.j. – novčanih jedinica

d.d. – dioničko društvo

7. POPIS LITERATURE

Knjige

[1] Neralić, L.; Šego, B.; 2009., Matematika, Zagreb, KIKA-GRAF, str. 287-288.

Internet

[2] http://maxima.sourceforge.net/, 26.05.2014.

[3] http://maxima.sourceforge.net/docs/manual/en/maxima.html, 27.05.2014.

[4] http://www.businessdictionary.com/definition/cost-function.html, 27.05.2014.

[5] http://lumens.fthm.hr/edata/2011/c04b8c97-7b7b-4bd8-bb5b-b196399d2433.pdf, 28.05.2014.

[6] http://www.megatrend-online.com/poslovna%20matematika/lekcije/lekcija19.htm, 28.05.2014.

[7] http://ef.sve-mo.ba/ljetna_skola/materijali/matematika/Elasticnost_Mostar.pdf, 28.05.2014.

8. POPIS PRILOGA