

Izrada aplikacije za kvadratnu jednadžbu

Juzbašić, Martin

Undergraduate thesis / Završni rad

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:912022>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-01-09**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA

Stručni studij

IZRADA APLIKACIJE ZA KVADRATNU
JEDNADŽBU

Završni rad

Martin Juzbašić

Osijek, 2016.

Obrazac Z1S: Obrazac za imenovanje Povjerenstva za obranu završnog rada na stručnom studiju

Osijek,

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za obranu završnog rada na stručnom studiju

Ime i prezime studenta:	Martin Juzbašić
Studij, smjer:	Stručni studij elektrotehnike, smjer Informatika
Mat. br. studenta, godina upisa:	A3399, 2007.
Mentor:	Doc.dr.sc. Tomislav Rudec - docent
Sumentor:	Doc.dr.sc. Alfonzo Baumgartner - docent
Predsjednik Povjerenstva:	
Član Povjerenstva:	
Naslov završnog rada:	Izrada aplikacije za kvadratnu jednadžbu
Primarna znanstvena grana rada:	
Sekundarna znanstvena grana (ili polje) rada:	
Zadatak završnog rada	Zadatak ovog završnog rada je matematički obraditi kvadratnu jednadžbu i izraditi aplikaciju koja za unesenu kvadratnu jednadžbu računa i prikazuje njezina svojstva.
Prijedlog ocjene pismenog dijela ispita (završnog rada):	
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: Postignuti rezultati u odnosu na složenost zadatka: Jasnoća pismenog izražavanja: Razina samostalnosti:
Potpis sumentora:	Potpis mentora:
Dostaviti:	
1. Studentska služba	
U Osijeku, 28. lipnja 2016. godine	Potpis predsjednika Odbora:



ETFOS
ELEKTROTEHNIČKI FAKULTET OSIJEK



Sveučilište Josipa Jurja Strossmayera u Osijeku

IZJAVA O ORIGINALNOSTI RADA

Osijek, 28. lipnja 2016.

Ime i prezime studenta:

Martin Juzbašić

Studij :

Stručni studij elektrotehnike, smjer Informatika

Mat. br. studenta, godina upisa:

A3399

Ovom izjavom izjavljujem da je rad pod nazivom:
Izrada aplikacije za kvadratnu jednadžbu

izrađen pod vodstvom mentora

Doc.dr.sc. Tomislav Rudec - docent

i sumentora

Doc.dr.sc. Alfonso Baumgartner - docent

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

Sadržaj

1. UVOD	1
2. MATEMATIČKA OBRADA KVADRATNE JEDNADŽBE	2
2.1 Rješavanje kvadratne jednadžbe	2
2.2 Svojstva kvadratne jednadžbe	3
2.3 Crtanje grafa kvadratne funkcije	5
2.4 Međusobni odnos dviju parabola	6
2.5 Primjena kvadratne funkcije	10
3. PREGLED KORIŠTENIH ALATA ZA IZRADU APLIKACIJE	11
3.1 Visual Studio	11
3.2 Microsoft Visual C#	13
4. IZRADA APLIKACIJE	14
4.1 Objašnjenje programskog dijela	17
5. ZAKLJUČAK	23
LITERATURA	24
SAŽETAK	25
ŽIVOTOPIS	26
PRILOZI	27

1. UVOD

Kvadratna jednadžba dio je elementarne algebre te se s njom učenici susreću u drugom razredu srednje škole. Njezina svojstva i primjena nalaze se u mnogim dijelovima matematici srodnih znanosti, ljudskih djelatnosti i života općenito.

S obzirom na to, postoji velika potreba za gotovim programom koji računa i prikazuje svojstva kvadratne jednadžbe. Dobro napisan i testiran program koji rješava kvadratnu jednadžbu ubrzava proces rješavanja kompleksnijih problema te eliminira mogućnost pogreške pri "ručnom" računanju. Služi i kako bi se brzo provjerilo rješenje pri rješavanju zadataka. U ovom je radu kvadratna jednadžba, njezina svojstva i primjena obrađena matematički. Na temelju toga, izrađena je aplikacija za računalo koja na osnovu unesene kvadratne jednadžbe računa njezina svojstva - nultočke i njihovu prirodu, diskriminantu, koordinate tjemena te prikazuje graf funkcije opisane zadanom kvadratnom jednadžbom. Aplikacija ima mogućnost prikaza grafa dvije unesene kvadratne jednadžbe i računanje koordinata njihova sjecišta.

Rad se sastoji iz tri dijela. U prvom je dijelu dana matematička obrada kvadratne jednadžbe i njezina primjena. Drugi dio opisuje alate upotrijebljene za izradu aplikacije. U trećem dijelu opisana je sama realizacija izrade aplikacije. U zaključku je definirano koliko je uspješno realizirana aplikacija s obzirom na zadane ciljeve.

1.1 Zadatak završnog rada

Cilj ovog završnog rada je matematički obraditi kvadratnu jednadžbu i izraditi aplikaciju koja za unesene kvadratne jednadžbe računa i prikazuje njihova svojstva.

2. MATEMATIČKA OBRADA KVADRATNE JEDNADŽBE

Kvadratna jednadžba je svaka jednadžba oblika:

$$ax^2 + bx + c = 0$$

pri čemu je x nepoznanica, a realni brojevi a , b i c predstavljaju koeficijente kvadratne jednadžbe [1].

a je vodeći (kvadratni) koeficijent koji se nalazi uz nepoznanicu x najvećeg (drugog) reda. Pri tome vrijedi uvijet $a \neq 0$. Kada bi vrijedilo $a = 0$, tada jednadžba ne bi bila kvadratna već linearna. b je linearni koeficijent uz nepoznanicu x prvog reda, a c predstavlja slobodni koeficijent.

Ovakva se jednadžba naziva kvadratnom ili jednadžbom drugog stupnja jer je najveći stupanj nepoznanice dva. Stoga ovakva jednadžba ima dva rješenja koja su ili oba realna ili konjugirano-kompleksna.

2.1 Rješavanje kvadratne jednadžbe

Svaki x koji zadovoljava kvadratnu jednadžbu naziva se rješenje ili korijen kvadratne jednadžbe. Kvadratna se jednadžba može riješiti faktorizacijom, nadopunom do potpunog kvadrata ili, najčešće, uporabom formule za izračun rješenja kvadratne jednadžbe.

Faktorizacija

Kvadratnu jednadžbu $ax^2 + bx + c = 0$ moguće je izraziti kao umnožak:

$$(px + q)(rx + s) = 0$$

pri čemu su rješenja ove jednadžbe:

$$x_1 = \frac{-q}{p}, x_2 = \frac{-s}{r}$$

U nekim je slučajevima, kada su $b = 0$ ili $c = 0$, lako faktorizirati kvadratnu jednadžbu. U drugim slučajevima, koriste se Vieteove formule:

$$x_1 + x_2 = \frac{-b}{a}, x_1 x_2 = \frac{c}{a}$$

Rješavanjem sustava ove dvije linearne jednadžbe dolazi se do rješenja x_1 i x_2 te se jednadžba faktorizira na način:

$$(x - x_1)(x - x_2) = 0$$

Nadopuna do potpunog kvadrata

Ovaj postupak se provodi uporabom algebarskog identiteta kvadrata binoma [1]:

$$x^2 + 2px + p^2 = (x + p)^2$$

na sljedeći način:

1. $ax^2 + bx + c = 0$

2. $x^2 + \frac{b}{a}x + \frac{c}{a} = 0$ početna jednadžba dijeli se s vodećim koeficijentom:

3. $x^2 + \frac{b}{a}x = -\frac{c}{a}$ oduzima se konstanta

4. $x^2 + \frac{b}{a}x + \left(\frac{b}{2a}\right)^2 = -\frac{c}{a} + \left(\frac{b}{2a}\right)^2$ dodaje se kvadrirana polovica koeficijenta uz x

5. $\left(x + \frac{b}{2a}\right)^2 = -\frac{c}{a} + \left(\frac{b}{2a}\right)^2$ lijeva strana se svodi na potpuni kvadrat

6. $\left(x + \frac{b}{2a}\right) = \pm\sqrt{-\frac{c}{a} + \left(\frac{b}{2a}\right)^2}$ korijenuju se obje strane jednadžbe

7. $x = -\frac{b}{2a} \pm \sqrt{-\frac{c}{a} + \left(\frac{b}{2a}\right)^2}$ konačno se dobiju dva rješenja.

Uporaba formule

Kako se ne bi svaki puta ponavljao postupak svodenja na potpuni kvadrat, češće se koristi gotova formula dobivena u gore navedenom postupku.

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

2.2 Svojstva kvadratne jednadžbe

Nekoliko je parametara koji govore o svojstvima kvadratne jednadžbe i kvadratne funkcije, a to su nultočke, diskriminanta i tjeme (ekstrem).

Nultočke

Nultočke, korijeni ili rješenja kvadratne jednadžbe predstavljaju vrijednosti $x_{1,2}$ koje zadovoljavaju kvadratnu jednadžbu. U grafu kvadratne funkcije to su točke u kojima je vrijednost funkcije jednaka nuli, odnosno točke u kojima funkcija sječe x-os.

Diskriminanta

U formuli za rješavanje kvadratne jednadžbe, izraz ispod korijena naziva se diskriminanta te se označava slovom D ili Δ .

$$D = b^2 - 4ac$$

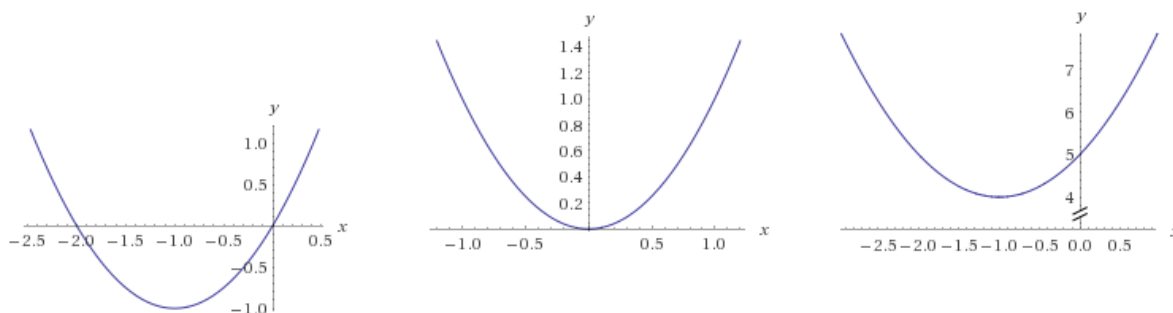
Diskriminanta ima veliku ulogu u određivanju prirode rješenja. Postoje tri mogućnosti[1]:

$D > 0$ - tada postoje dva realna različita rješenja.

$D = 0$ - tada postoji jedno dvostruko realno rješenje.

$D < 0$ - tada postoje dva konjugirano kompleksna rješenja.

Grafički su rješenja prikazana sjecištem grafa kvadratne funkcije i x-osi kao što je prikazano na slici 2.1.



Slika 2.1: Ovisnost rješenja o diskriminanti, s lijeva nadesno $D > 0$, $D = 0$, $D < 0$

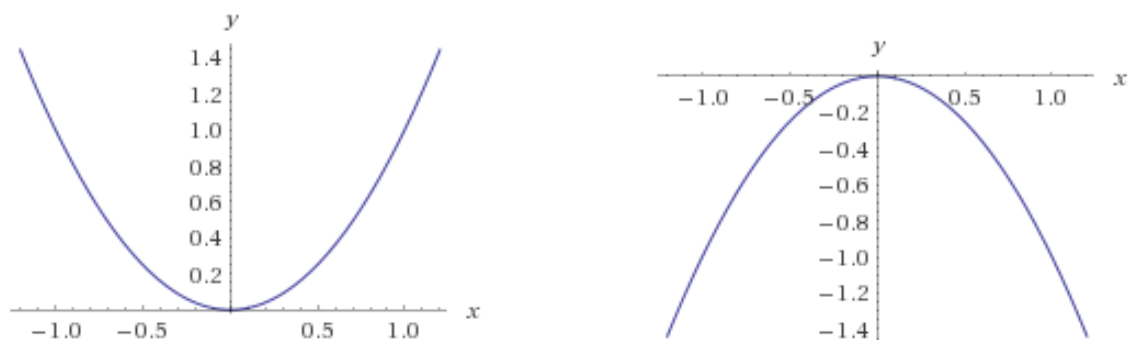
Uloga vodećeg koeficijenta kod određivanja ekstrema

S obzirom da je kvadratna funkcija polinom drugog stupnja, postoji jedan ekstrem. On može biti maksimum - u kojem je vrijednost funkcije najveća ili minimum - u kojem je vrijednost funkcije najmanja. Ekstrem kvadratne funkcije naziva se tjeme. Priroda ekstrema ovisi o vodećem koeficijentu.

Ako je $a > 0$, tjeme je minimum.

Ako je $a < 0$, tjeme je maksimum.

Na slici 2.2 prikazana je ovisnost otvora parabole o vodećem koeficijentu.



Slika 2.2: Ovisnost otvora parabole o vodećem koeficijentu, s lijeva nadesno: $a > 0$, $a < 0$

2.3 Crtanje grafa kvadratne funkcije

Funkcija $f(x) = ax^2 + bx + c$ naziva se kvadratna funkcija. Graf kvadratne funkcije je parabola. Položaj parabole u koordinatnom sustavu, njezina veličina i otvorenost ovise o koeficijentima a, b i c . Ukoliko je $a > 0$, tada je parabola otvorena prema gore te je njezino tjeme minimum. Ukoliko je $a < 0$, tada je otvor prema dolje i tjeme je maksimum. Koordinate tjemena $T(x_T, y_T)$ nalaze se na koordinatama:

$$x_T = \frac{-b}{2a}, y_T = \frac{-D}{4a}$$

Nultočke ili rješenja kvadratne jednadžbe odgovaraju x-koordinatama točaka u kojima funkcija sječe x-os. Ako parabola sječe x-os tada postoje dva realna rješenja, ako tjeme parabole dira x-os tada postoji jedno dvostruko realno rješenje, a ako parabola ne sječe niti ne dira x-os, tada su rješenja konjugirano kompleksna.

Uobičajeno crtanje grafa kvadratne funkcije provodi se po sljedećim koracima:

1. Određuje se predznak vodećeg koeficijenta te se zaključuje o otvorenosti parabole.
2. Računaju se nultočke.
3. Određuju se koordinate tjemena.
4. Po potrebi se odrede još neke točke na grafu.
5. Crta se graf funkcije.

2.4 Međusobni odnos dviju parabola

Parabole u koordinatnom sustavu mogu biti u četiri različita međuosobna; mogu se mimoilaziti, dodirivati, presjecati u jednoj ili dvije točke ili potpuno preklapati [2].

Postupak određivanja sjecišta

Pretpostavljaju se dvije kvadratne funkcije:

$$f(x) = ax^2 + bx + c$$

$$g(x) = ex^2 + fx + g$$

Mjesta na kojima se ove dvije parabole sijeku nazivaju se sjecišta. U slučaju kada parabole imaju presjek tada postoje koordinate (x,y) koje pripadaju objema parabolama. Ovaj se problem svodi na rješavanje sustava dvije jednačbe s dvije nepoznanice, x i y . Prema tome, mogu se izjednačiti njihove y koordinate te se traže odgovarajuće koordinate x kao što je prikazano sljedećim postupkom.

1. $f(x) = g(x)$, započinje se izjednačavanjem funkcija
2. $ax^2 + bx + c = ex^2 + fx + g$, uvrste se izrazi koji odgovaraju y koordinatama
3. $(a - e)x^2 + (b - f)x + (c - g) = 0$, cijeli se izraz prebacuje na jednu stranu te se dobiva poznati oblik kvadratne jednačbe

U koraku 3 je moguće dobiti linearnu ili kvadratnu jednačbu. Ukoliko je jednačba linearna, tj. ukoliko $a = e$, tada se za dobiveno rješenje x , uvrštavanjem u neku od početnih jednačbi dobiva koordinata y . Parabole se tada sijeku u jednoj točki. Ukoliko je slučaj takav da je $a = e$ i $b = f$, a $c \neq g$, tada ne postoji sjecište jer dobivena jednačba nije istinita, npr $3 = 0$. U slučaju da su svi odgovarajući koeficijenti jednaki, radi se o istim parabolama koje se potpuno preklapaju.

U slučaju da je dobivena jednačba kvadratna, prelazi se na sljedeći korak:

4. $x_{1,2} = \frac{-(b - f) \pm \sqrt{(b - f)^2 - 4(a - e)(c - g)}}{2(a - e)}$, upotrebljava se formula za kvadratnu jednačbu

5. $y_1 = ax_1^2 + bx_1 + c$, $y_2 = ax_2^2 + bx_2 + c$, za dobivene koordinate x , uvrštavanjem u jednu od funkcija dobivaju se odgovarajuće koordinate y .

Kao i svaka, i ovako dobivena kvadratna jednačba može imati dva realna, jedno dvostruko

ili dva kompleksno-konjugirana rješenja. O tome ovisi i koliko će sjecišta postojati. Brzi način za određivanje broja sjecišta je izračun diskriminante dobivene kvadratne jednadžbe:

$$D = (b - f)^2 - 4(a - e)(c - g)$$

Razlikuju se tri slučaja:

$D > 0$, postoje dva sjecišta (presjecaju se)

$D = 0$, postoji jedno sjecište (dodiruju se)

$D < 0$, nema sjecišta (mimoilaze se).

Mimoilaženje

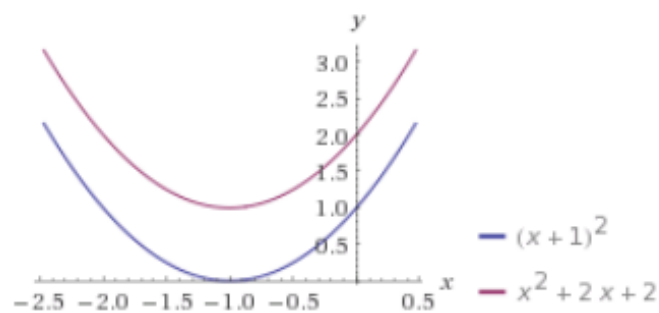
Parabole se mimoilaze ako nemaju niti jednu točku zajedničku. Rješavanjem ovog sustava jednadžbi dobiva se jednadžba koja nema rješenje ili kvadratna jednadžba čija je diskriminanta negativna.

Primjer 1:

$$f(x) = x^2 + 2x + 1$$

$$g(x) = x^2 + 2x + 2$$

Rješavanjem sustava ove dvije jednadžbe, dobiva se $-1 = 0$, jednadžba nema rješenje te se parabole mimoilaze kao što je prikazano slikom 2.3.



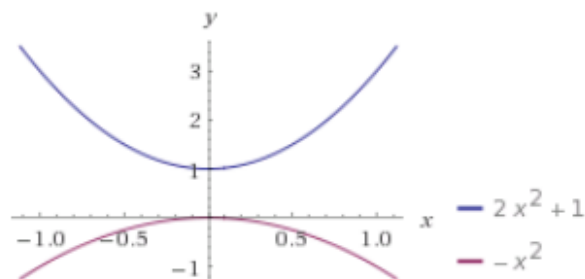
Slika 2.3: Primjer mimoilaženja parabola

Primjer 2:

$$f(x) = 2x^2 + 1$$

$$g(x) = -x^2$$

Jednadžba dobivena rješavanjem sustava je kvadratna pri čemu je $D = -12$. Parabole se mimoilaze kao što je prikazano slikom 2.4.



Slika 2.4: Primjer mimoilaženja parabola

Dodirivanje

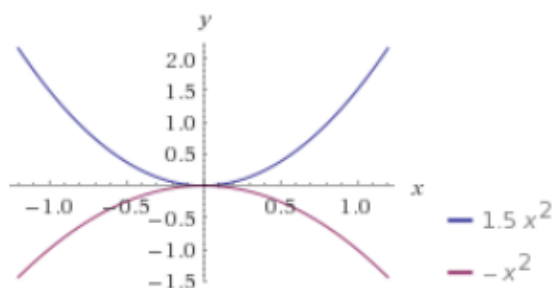
Parabole se dodiruju ako je jednadžba dobivena rješavanjem sustava kvadratna i njezina je diskriminanta jednaka nuli. Parabole se dodiruju ako postoji samo jedna točka koja pripada objema parabolama, a da se pritom one ne presjecaju.

Primjer 3:

$$f(x) = 1.5x^2$$

$$g(x) = -x^2$$

Diskriminanta dobivene kvadratne jednadžbe $D = 0$ te se parabole dodiruju u točki $(0,0)$ kao na slici 2.5.



Slika 2.5: Primjer dodira parabola

Presjecanje

Parabole se sijeku u jednoj točki ako je jednadžba dobivena rješavanjem sustava linearna te

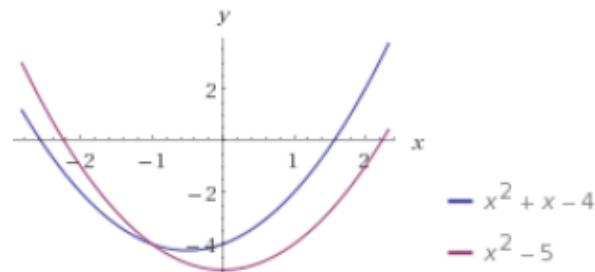
ima jedno rješenje ili u dvije točke ako je jednadžba kvadratna i ima dva realna rješenja.

Primjer 4:

$$f(x) = x^2 + x - 4$$

$$g(x) = x^2 - 5$$

Rješenje sustava je linearna jednadžba čije je rješenje $x = -1$. Uvrštavanjem u jednu od početnih jednadžbi, dobiva se $y = -4$. Parabole se sijeku u jednoj točki $(-1, -4)$ kao što je prikazano slikom 2.6.



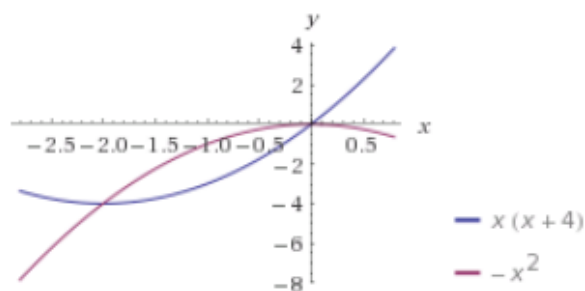
Slika 2.6: Primjer presjeka parabola u jednoj točki

Primjer 5:

$$f(x) = x^2 + 4x$$

$$g(x) = -x^2$$

Rješenje sustava je kvadratna jednadžba čija je diskriminanta veća od nule, $D = 16$. Traženjem rješenja dobivene kvadratne jednadžbe, dobivaju se $x_1 = -2$, $x_2 = 0$. Odgovarajuće y koordinate su $y_1 = -4$ i $y_2 = 0$. Parabole se sijeku u dvije točke $(-2, -4)$ i $(0, 0)$ kao što je prikazano slikom 2.7.



Slika 2.7: Primjer presjeka parabola u dvije točke

Potpuno preklapanje

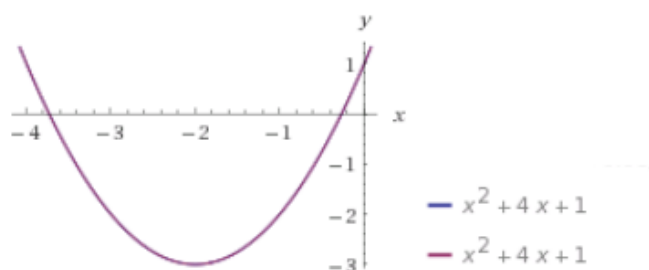
Ovo je poseban slučaj međusobnog odnosa dviju parabola. Zapravo se radi o istim parabolama, što znači da se parabole potpuno preklapaju kada su im odgovarajući koeficijenti jednaki. U tom slučaju sve točke koje leže na parabolama su točke sjecišta.

Primjer 6:

$$f(x) = x^2 + 4x + 1$$

$$g(x) = x^2 + 4x + 1$$

Kao što je prikazano slikom 2.8, parabole se potpuno preklapaju.



Slika 2.8: Primjer preklapanja parabola

2.5 Primjena kvadratne funkcije

Kvadratna funkcija i njezin graf nalaze se u mnogim područjima znanosti i inženjerstva. Oblik parabole može opisivati mnoge pojave u prirodi poput gibanja tijela izbačenog horizontalno ili pod kutom. Poznati su mnogi građevinski objekti poput mostova ili lukova paraboličnog oblika. Ovakav oblik pojavljuje se i u poslovnim izračunima profita i u raznim drugim područjima života. Kvadratna se funkcija koristi u izračunu kvadratnih površina i drugih geometrijskih problema kao i u statističkim analizama.

3. PREGLED KORIŠTENIH ALATA ZA IZRADU APLIKACIJE

Aplikacija za rješavanje i prikaz kvadratne jednadžbe napisana je u razvojnom okruženju Microsoft Visual Studio 2013., u programskom jeziku Visual C#.

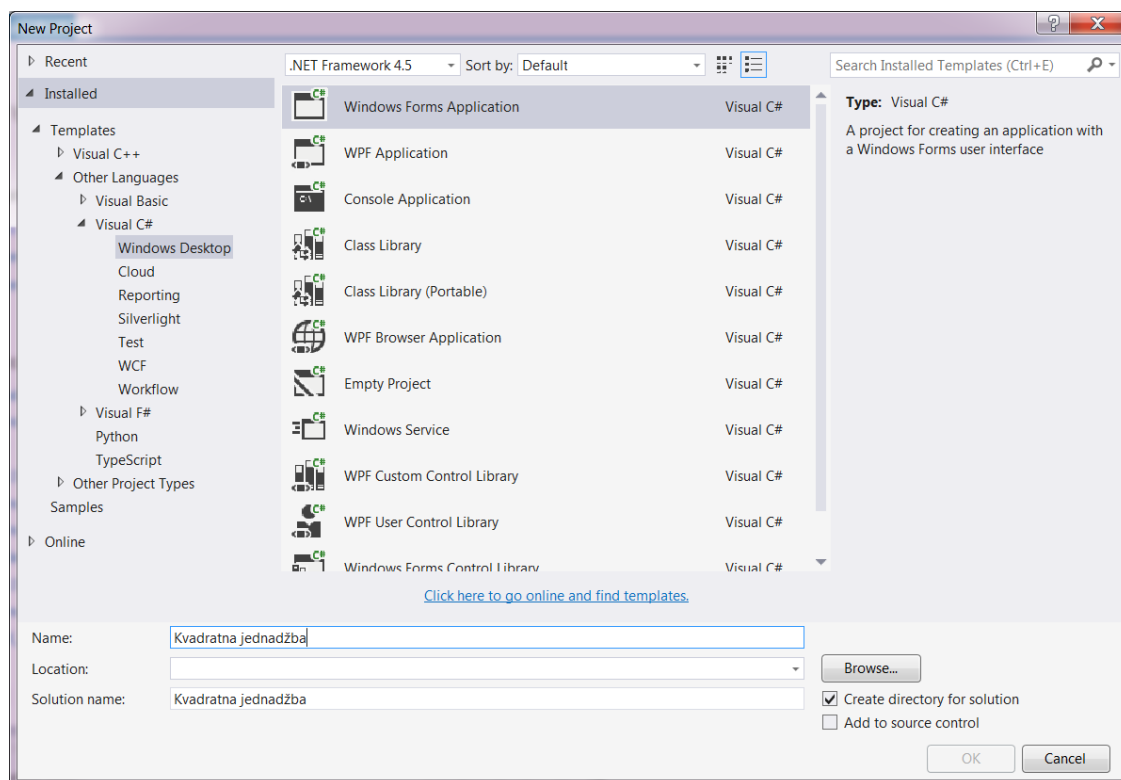
3.1 Visual Studio

Visual Studio je integrirano razvojno okruženje za razvoj aplikacija za Windows, Android, iOS te aplikacija za internet i usluge u oblaku računala [3]. Visual Studio koristi Microsoftove razvojne platforme poput Windows API, Windows Forms itd. U Visual Studio uključen je editor koda kao i prevoditelj. U njemu se nalaze alati za razvoj aplikacija s grafičkim sučeljem kao i alati za web-dizajn i baze podataka.

U Visual Studiu podržani su mnogi programski jezici poput C, C++, C#, Python, Ruby, Node.js kao i XML/XSLT, HTML/YXHTML, JavaScript, CSS i njihovi prevoditelji.

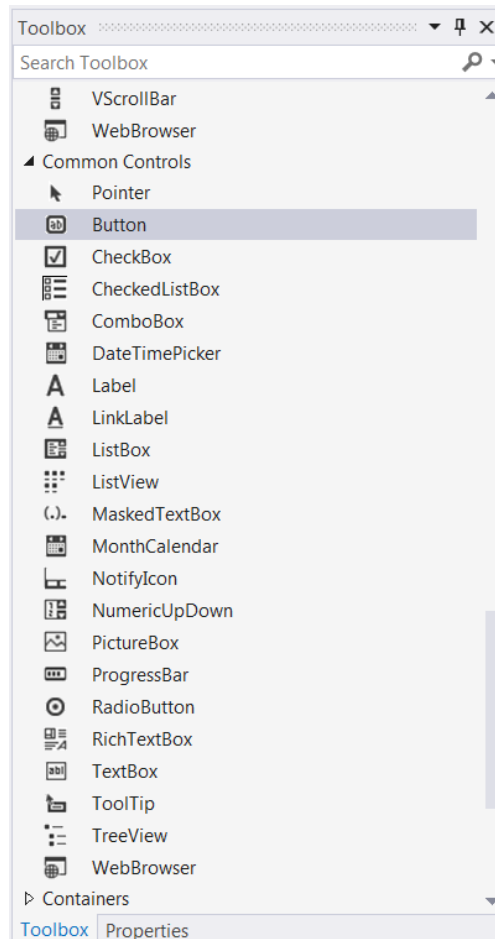
Windows Forms Designer

Windows forme koriste se za razvoj grafičkog sučelja. Odabir izrade Windows forme obavlja se pri stvaranju novog projekta u Visual Studiu kao što je prikazano slikom 3.1.



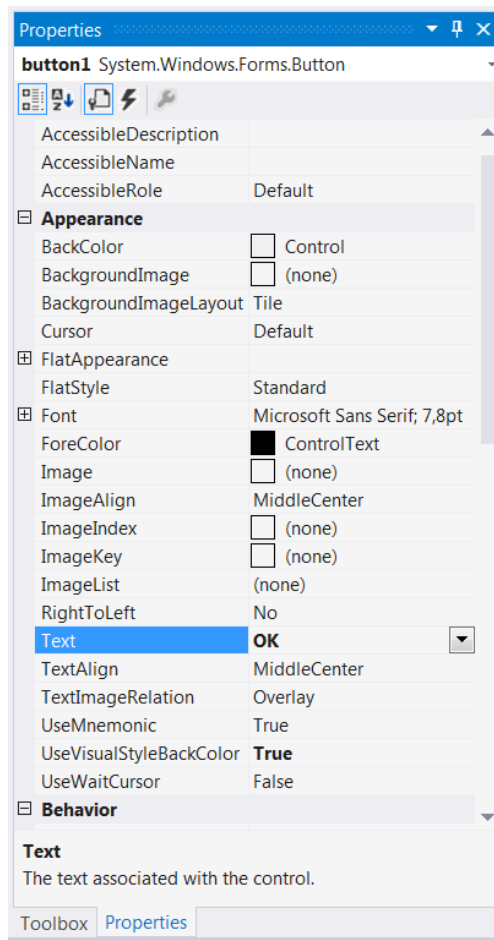
Slika 3.1: Prikaz sučelja pri izboru rada s Windows formama

Forma predstavlja prozor aplikacije koji korisnik vidi te u njemu odabire željene funkcije programa. U jednoj aplikaciji moguće je postaviti više formi koje će se otvarati ovisno o tome što korisnik želi pokrenuti unutar aplikacije. Izgled aplikacije moguće je kreirati prevlačenjem željene kontrole za prikaz podataka, npr. gumba, labele, izbornika ili tekst editora iz alatne trake prikazane na slici 3.2 direktno u prozor aplikacije.



Slika 3.2: Prikaz alata s kontrolama

Pozicija i veličina gumba ili labele odabire se smještanjem i razvlačenjem u samoj formi uporabom miša. Na ovaj način programeru se olakšava sam dizajn grafičkog sučelja te se usmjerava na razvoj logike programa. Ostala svojstva svakog gumba (boja, font...), labele i ostalih kontrola za prikaz mijenjaju se odabirom željene vrijednosti u njihovom izborniku poput onog na slici 3.3.



Slika 3.3: Odabir svojstava

Korisničko sučelje povezano je s kodom uporabom programskog modela uzrokovanog događajem. To znači da dvostrukim klikom na gumb ili labelu se otvara njezin dio u kodu. Programiranjem se izabire što će se dogoditi ako korisnik klikne na gumb ili odabere neku opciju iz izbornika itd. To uključuje otvaranje novih formi, zatvaranje prozora, izlaz iz aplikacije, računске operacije, prikaz podataka, slika, promjenu podataka te mnoge druge mogućnosti.

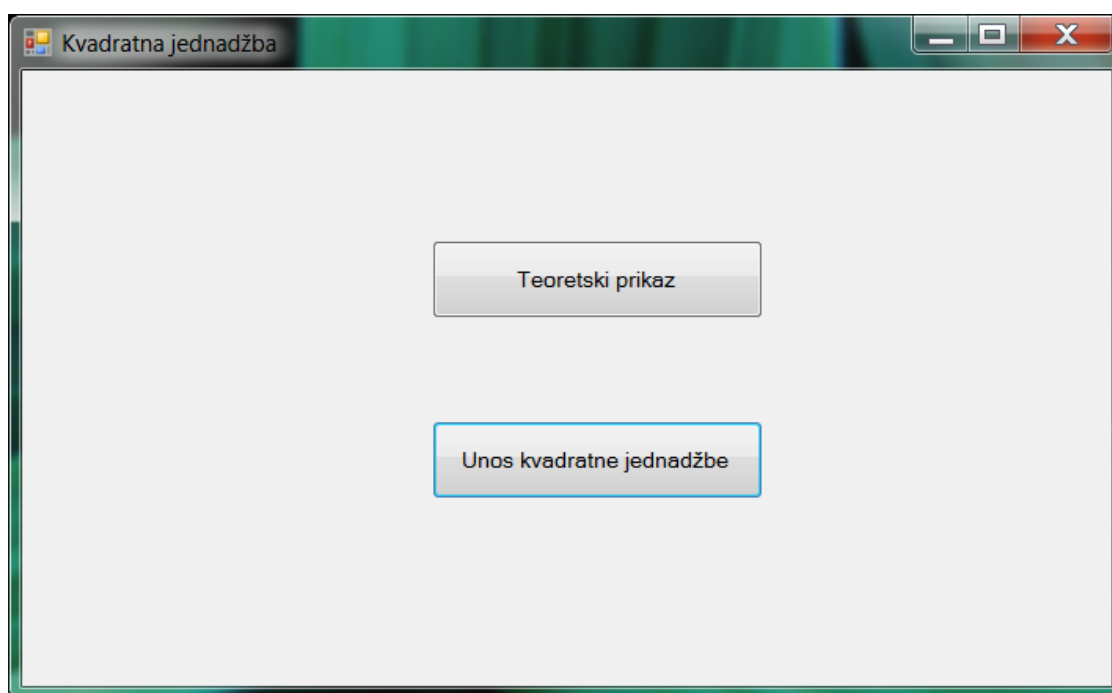
3.2 Microsoft Visual C#

C# je objektno orijentiran programski jezik razvijen od Microsofta unutar .NET inicijative. Svrha mu je da bude jednostavan i moderan, višenamjenski jezik. Sintaksom je sličan C-u. Microsoft Visual C# je implementacija C# jezika unutar Visual Studia usmjeren na .NET Framework. Prevoditelj je neovisan o .NET Framework, dok su usluge programskog jezika vezane uz Visual Studio. Visual C# podržava Visual Studio Class dizajn, Data dizajn i ono najvažnije, dizajn Formi.

4. IZRADA APLIKACIJE

Zamisao je bila izraditi aplikaciju koja će za unesenu kvadratnu jednadžbu izračunati i prikazati njezine nultočke, diskriminantu, koordinate tjemena te u koordinatni sustav smjestiti zadanu kvadratnu funkciju. Nadalje, ukoliko korisnik želi, može unijeti koeficijente još jedne kvadratne jednadžbe te dobiti prikaz obje funkcije u koordinatnom sustavu kao i koordinate njihova sjecišta, ukoliko ono postoji.

U Visual Studio otvoren je novi projekt za izradu desktop aplikacije uporabom Windows formi i Visual C#. Program se sastoji iz tri forme. Prva forma se otvara pri pokretanju aplikacije te je njezin izgled prikazan na slici 4.1.



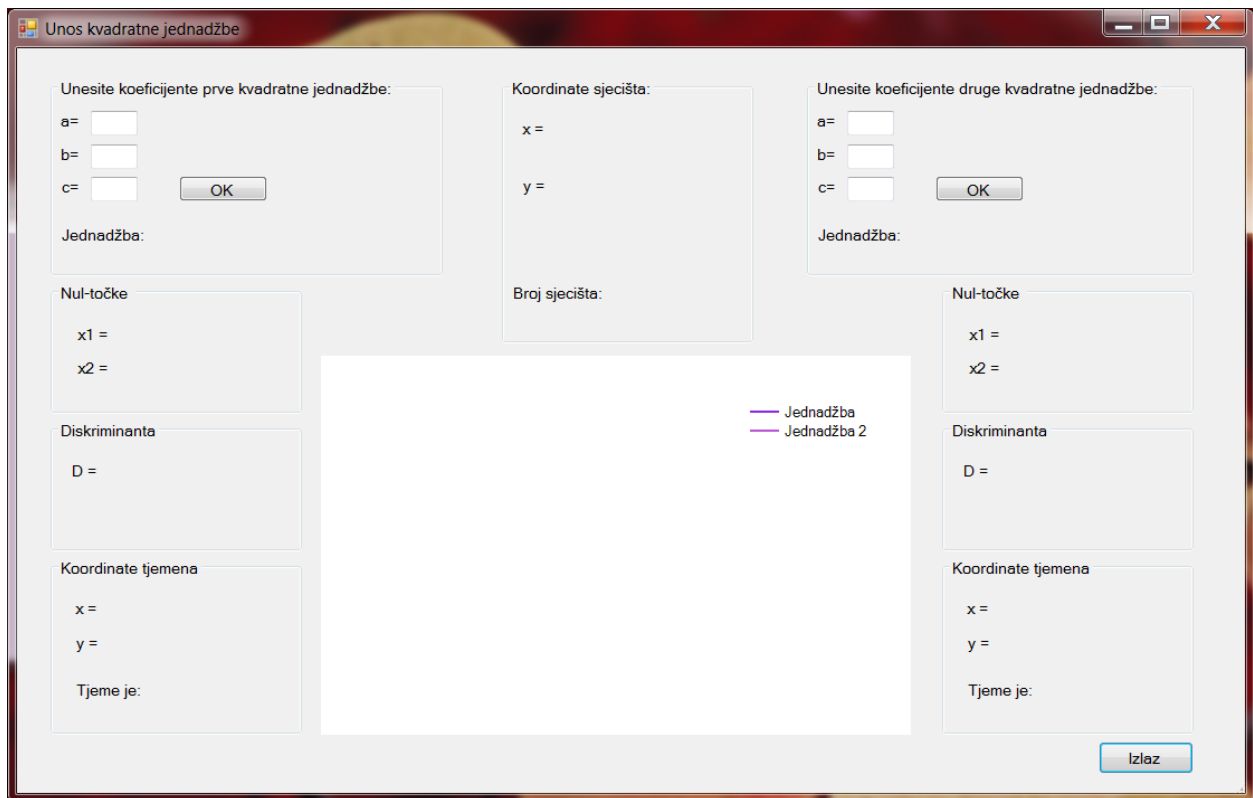
Slika 4.1: Prikaz forme pri pokretanju aplikacije

Klikom na gumb Teoretski prikaz otvara se druga forma čiji je prikaz na slici 4.2. Ovdje se nalazi teorijska osnova kvadratne jednadžbe čiji je cilj upoznati korisnika s matematičkom podlogom kako bi znao tumačiti rezultate dobivene nakon unosa kvadratne jednadžbe.

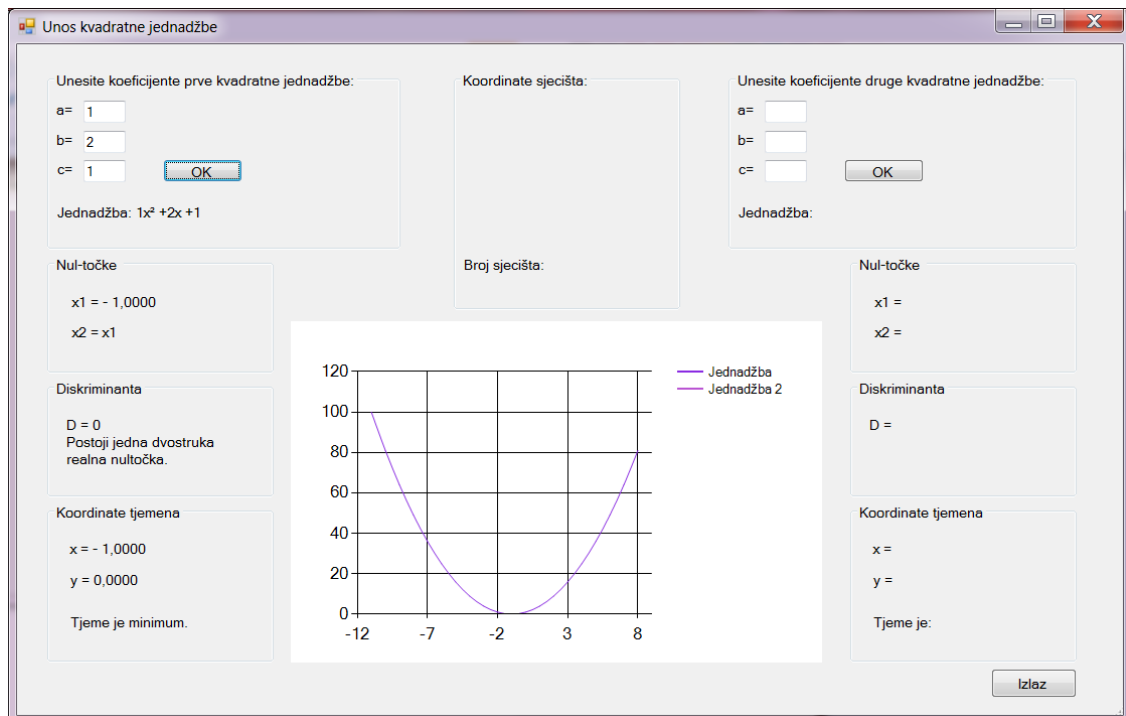
KVADRATNA JEDNADŽBA		
$ax^2+bx+c=0$		
<p>Koeficijenti:</p> <p>a – vodeći b – linearni c – slobodni</p> <p>$a \neq 0$</p>	<p>Nultočke:</p> <p>-rješenja kvadratne jednadžbe -postoje dvije nultočke -dobivaju se formulom</p> $x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ <p>Vieteove formule:</p> $x_1 + x_2 = \frac{-b}{a}, x_1 \cdot x_2 = \frac{c}{a}$	<p>Diskriminanta:</p> $D = b^2 - 4ac$ <p>D > 0 dva realna rješenja</p> <p>D = 0 jedno dvostruko realno rješenje</p> <p>D < 0 dva konjugirano kompleksna rješenja</p>
<p>Tjeme:</p> <p>a > 0 minimum</p> <p>a < 0 maksimum</p> $x_T = \frac{-b}{2a}$ $y_T = \frac{-D}{4a}$	<p style="text-align: center;">Graf:</p>	

Slika 4.2: Prikaz druge forme s teorijskom podlogom

Klikom na gumb Unos kvadratne jednadžbe otvara se treća forma, prozor kao na slici 4.3. Tu se očekuje od korisnika da unese koeficijente kvadratne jednadžbe. Klikom na gumb OK u lijevom dijelu prozora, prikazuju se podaci dobiveni računom u pozadini programa te se iscertava graf zadane kvadratne funkcije. Primjer je prikazan na slici 4.4.



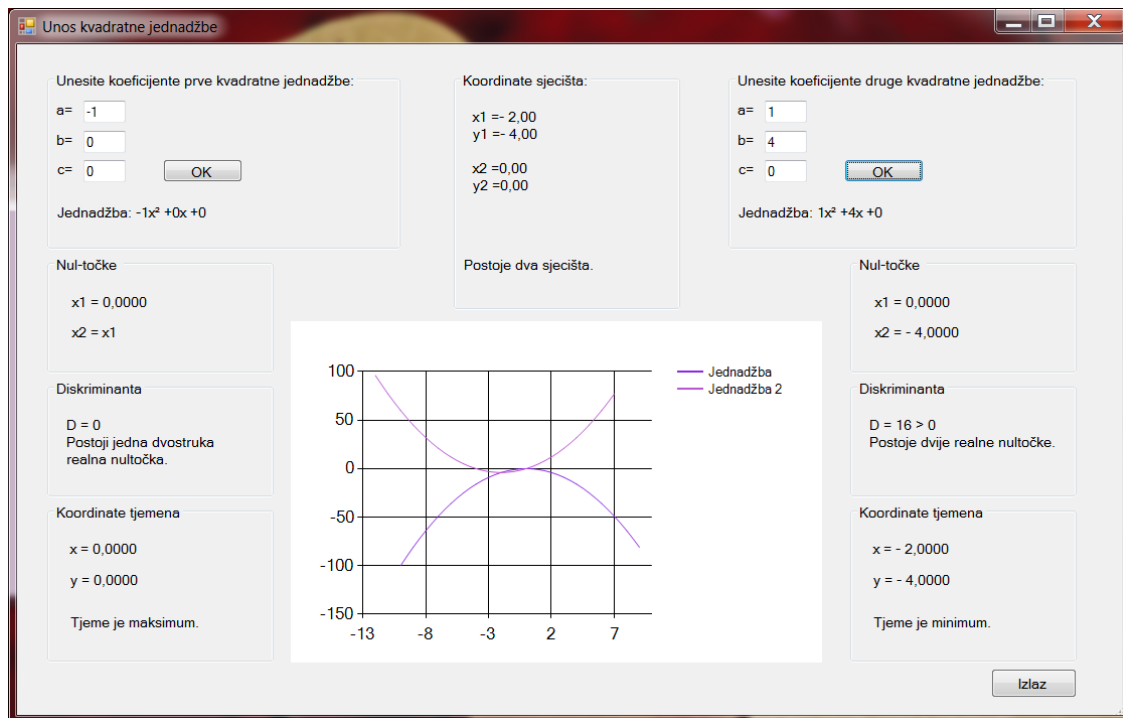
Slika 4.3: Prikaz forme u kojoj korisnik unosi kvadratnu jednadžbu



Slika 4.4: Prikaz forme nakon unosa jedne željene kvadratne jednadžbe

Ukoliko korisnik želi unijeti još jednu kvadratnu jednadžbu te dobiti međusobni odnos tih dviju parabola kao i koordinate sjecišta ukoliko ono postoji, treba unijeti koeficijente druge

kvadratne jednadžbe te pritisnuti gumb OK u desnom dijelu prozora. Kako to izgleda nakon programske obrade prikazano je primjerom na slici 4.5.



Slika 4.5: Prikaz forme nakon unosa dvije željene kvadratne jednadžbe

4.1 Objašnjenje programskog dijela

Izrada aplikacije svodi se na odabir dizajnerskih komponenti (labele, dugmad, text-box) te programiranja željene akcije uslijed klika na gumb. U prvoj formi nalaze se dva gumba te klikom na svaki gumb otvara se posebna forma. Treća je forma programski najzahtjevnija jer se programira logika izračuna nul-točaka, diskriminante, koordinata tjemena i grafičkog prikaza kvadratne jednadžbe te koordinata sjecišta s drugom zadanom kvadratnom jednadžbom.

U formi se nalaze dva gumba OK. Lijevi OK pretpostavlja da korisnika zanima samo jedna kvadratna jednadžba, a desni OK u obzir uzima obje jednadžbe te ih prikazuje, računa njihova svojstva i njihovo sjecište. Svojstva lijeve i desne kvadratne jednadžbe računaju se na isti način te su pojedini dijelovi objašnjeni u nastavku. Cjelokupan kod nalazi se u priložima.

Unos koeficijenata

Korisnik zatiče prazna polja kao na slici 4.3 te unosi željene koeficijente. Koeficijenti moraju biti realni brojevi uz uvijet da $a \neq 0$ kako bi jednadžba bila kvadratna. U suprotnom, aplikacija traži ponovni unos koeficijenata. U kodu, ovaj dio riješen je na sljedeći način kako bi se sadržaj prozora u koji je unešen koeficijent mogao koristiti kao realna varijabla:

```

double a,b,c;
bool isDoublea = double.TryParse(textBox1.Text, out a);
    if (isDoublea && a!=0) //a ne smije biti nula
    {
        bool isDoubleb = double.TryParse(textBox2.Text, out b);
        if(isDoubleb)
        {
            bool isDoublec = double.TryParse(textBox3.Text, out c);
            if(isDoublec)
            {
                ...
            }
        }
    }

```

Ispis jednadžbe

Klikom na gumb OK, ispisuje se jednadžba u obliku na koji je korisnik navikao: $ax^2 + bx + c$. Programski je ovo riješeno na sljedeći način:

```

label13.Text = "Jednadzba: " + a + "x\xB2 +" + b + "x +" + c ;

```

Jedini je manjak ovog zapisa što u slučaju negativnog koeficijenta ostaje i predznak + nakon čega dolazi predznak - (npr. $3x^2 + -2x + -1$).

Diskriminanta

S obzirom da se u aplikaciji navodi kakva je diskriminanta ($D > 0$, $D = 0$ ili $D < 0$) te ovisno o tome koliko je nul-točaka i kakva je njihova priroda, prvo se računa diskriminanta. To se odvija sljedećom linijom koda:

```

double D = ((b*b) - (4*a*c));

```

Nultočke

Ovisno o tome kakva je diskriminanta uvelo se uvjetno grananje te se nultočke računaju i ispisuju ovisno o diskriminanti. Ako postoji jedno realno rješenje, tj. ako je $D = 0$, tada je potrebno izračunati samo jednu nultočku uporabom formule za izračun kvadratne jednadžbe. Uslijed toga program na predviđeno mjesto u formi ispisuje tu nultočku te ispisuje da je to dvostruka realna nultočka.

```

//jedno realno rjesenje
    if (D == 0)
    {
        x1 = (-b + Math.Sqrt((b * b) - (4 * a * c))) / (2 * a);
        label5.Text = "x1 = " + x1.ToString("N4");
        label6.Text = "x2 = x1";
    }

```

```

        label9.Text = "D = " + D + " = 0 Postoji jedna
                    dvostruka realna nultočka.";
    }

```

U slučaju da je $D > 0$, računaju se dvije nultočke. Jedina je razlika što za drugu nultočku ispred korijena u formuli za izračun dolazi predznak minus. Također se ispisuju nultočke na predviđeno mjesto i prigodan tekst.

```

// dva realna rjesenja
    if (D > 0)
    {
        x1 = (-b + Math.Sqrt((b * b) - (4 * a * c))) / (2 * a);
        x2 = (-b - Math.Sqrt((b * b) - (4 * a * c))) / (2 * a);
        label5.Text = "x1 = " + x1.ToString("N4");
        label6.Text = "x2 = " + x2.ToString("N4");
        label9.Text = "D = " + D + " > 0 Postoje dvije realne
                    nultočke.";
    }

```

U slučaju $D < 0$, rješenja su konjugirano kompleksna. Kako programski jezik ne pozna aritmetiku kompleksnih brojeva, ovome se doskočilo podjelom broja na realni i imaginarni dio te ručnim upisivanjem imaginarne jedinice uz imaginarni dio broja. S obzirom da se u ovom slučaju ispod korijena u stvarnosti nalazi negativni broj, u programu je dodan predznak minus kako bi broj postao pozitivan te kako bi program mogao računati njegov korijen. Ponovno se nultočke ispisuju u aplikaciji uz prigodnu rečenicu.

```

//dva kompleksna rjesenja
    else if (D < 0)
    {
        double x1_r, x1_i, x2_r, x2_i;
        x1_r = -b / (2 * a); //realni dio rjesenja
        x1_i = Math.Sqrt(-((b*b) - (4*a*c)))/(2*a); //
                    imaginarni dio rjesenja
        x2_r = -b / (2 * a);
        x2_i = -1* Math.Sqrt(-((b * b) - (4 * a * c)))/(2*a);
        label5.Text = "x1 = " + x1_r.ToString("N2") + "+" +
                    x1_i.ToString("N2") + "i";
        label6.Text = "x2 = " + x2_r.ToString("N2") + x2_i.
                    ToString("N2") + "i";
        label9.Text = "D = " + D + " < 0 Postoje dvije
                    konjugirano kompleksne nultočke.";
    }

```

Koordinate tjemena

Prvo se računaju koordinate tjemena uporabom poznatih izraza te se ispisuje rečenica je li tjeme maksimum ili minimum ovisno o predznaku vodećeg koeficijenta.

```

//koordinate tjemena

```



```

double Tx, Ty;
Tx = -b / (2 * a);
Ty = -D / (4 * a);
label10.Text = "x = " + Tx.ToString("N4");
label11.Text = "y = " + Ty.ToString("N4");
if (a > 0) label12.Text = "Tjeme je minimum.";
else label12.Text = "Tjeme je maksimum.";

```

Koordinate tjemena, kao i nultočke zaokružene su na četiri decimalna mjesta kako bi ispis bio ljepši. Osim u slučaju kompleksnih nultočaka, tada su brojevi zaokruženi na dva decimalna mjesta.

Iscrtavanje grafa funkcije

Windows forme imaju u alatima predložak za crtanje grafa. Izabran je graf koji točke povezuje glatkom linijom. Prilikom pokretanja aplikacije, iz memorije se brišu spremljene točke kako prilikom višestrukog pokretanja aplikacije ne bi dolazilo do iscrtavanja pogrešnog seta točaka.

```

//brisanje tocaka iz memorije (zbog ponovnog pokretanja)
this.chart1.Series["Jednadzba"].Points.Clear();

```

Graf je potrebno prikazati tako da je uvijek prikazano tjeme parabole jer ono predstavlja njezinu os simetrije i intuitivno je da korisnik vidi taj dio grafa. Stoga je uporabom for petlje uzeta x-koordinata tjemena te su pridružene vrijednosti y za sve x udaljene za 10 lijevo i desno od tjemena. Koordinata x tjemena pretvorena je u cijeli broj kako se na x-osi ne bi pojavljivali decimalni brojevi što utječe na izgled grafa. U slučaju iscrtavanja grafova dvije funkcije, oni su prikazani tako da se oba uvijek vide bez obzira gdje se nalaze jedan u odnosu na drugoga. Taj dio nije programiran već je unaprijed postavljen u samom Visual Studiu.

```

//prikaz grafa
for (int x = (int)Tx - 10; x < (int)Tx + 10; x++)
{
    double y = (a * x * x) + (b * x) + c;
    this.chart1.Series["Jednadzba"].Points.AddXY(x, y);
}

```

Sjecište dviju parabola

Kada korisnik pritisne desni OK, funkcija računa sva svojstva za obje kvadratne jednadžbe te poziva funkciju za izračun koordinata sjecišta ukoliko ono postoji. S obzirom da se sjecište nalazi rješavanjem sustava jednadžbi, kod je podijeljen u dva dijela. Prvi dio se izvršava ukoliko je dobiveni sustav linearna jednadžba, a drugi dio ako je dobiveni sustav kvadratna jednadžba.

Ukoliko je dobivena jednadžba linearna, tada se odvija sljedeći dio koda. Postoji mogućnost da sustav nema rješenje, da ima jedno rješenje ili da ih ima bezbroj. Ovisno o tomu, ispisuje poruku da parabole nemaju presjek, da imaju jednu točku sjecišta ili da se preklapaju.

```
//nije kvadratna:
    if(a==d)
    {
        //nema rjesenja, mimoilazenje:
        if(b==g && c!=f)
        {
            label25.Text = "Nema presjeka.";
            label26.Text = " ";
            label27.Text = " ";
        }

        //bezbroj rjesenja, preklapanje:
        else if(b==g && c==f)
        {
            label25.Text = "Parabole su jednake i \
                npreklapaju se.";
            label26.Text = " ";
            label27.Text = " ";
        }

        //jedno rjesenje, jedno sjeciste:
        else if(b!=g)
        {
            double x1 = (f - c) / (b - g);
            double y1 = a * x1 * x1 + b * x1 + c;

            label25.Text = "x1 =" + x1.ToString("N2") + "\
                ny1 =" + y1.ToString("N2");
            label26.Text = " ";
            label27.Text = "Postoji jedno sjeciste.";
        }
    }
}
```

U slučaju da dobiveni sustav jednadžbi kao rješenje ima kvadratnu jednadžbu, tada je potrebno računati njezinu diskriminantu ovisno o kojoj se parabole mimoilaze ($D < 0$), dodiruju ($D = 0$) ili sjeku u dvije točke ($D > 0$).

```
//kvadratna:
double D = (b - g) * (b - g) - 4 * (a - d) * (c - f);
if (a != d)
{
    //presjecanje u dvije tocke, D>0:
```

```

if (D > 0)
{
    double x1 = (-(b - g) + Math.Sqrt(D)) / (2 * (a
        - d));
    double x2 = (-(b - g) - Math.Sqrt(D)) / (2 * (a
        - d));
    double y1 = a * x1 * x1 + b * x1 + c;
    double y2 = a * x2 * x2 + b * x2 + c;

    label25.Text = "x1 =" + x1.ToString("N2") + "\
        ny1 =" + y1.ToString("N2");
    label26.Text = "x2 =" + x2.ToString("N2") + "\
        ny2 =" + y2.ToString("N2");
    label27.Text = "Postoje dva sjecista.";
}

//dodirivanje, D=0
else if (D == 0)
{
    double x1 = (-(b - g) + Math.Sqrt(D)) / (2 * (a
        - d));
    double y1 = a * x1 * x1 + b * x1 + c;

    label25.Text = "x1 =" + x1.ToString("N2") + "\
        ny1 =" + y1.ToString("N2");
    label26.Text = " ";
    label27.Text = "Postoji jedno mjesto dodira.";
}

//mimoilazenje, D<0
else if (D < 0)
{
    label25.Text = "Nema presjeka.";
    label26.Text = " ";
    label27.Text = " ";
}
}
}
}

```

U formi postoji još i gumb Izlaz. Klikom na njega izlazi se iz trenutne forme te se korisnika vraća na početni zaslon prikazan slikom 4.1. Iz aplikacije se izlazi klikom na x u desnom gornjem uglu kao što je to uobičajeno u svim Windows aplikacijama.

5. ZAKLJUČAK

U ovom je radu izrađena aplikacija koja računa parametre unesene kvadratne jednadžbe i prikazuje graf kvadratne funkcije.

Bilo je potrebno temeljito proučiti matematički podlogu kako bi se razvilo ispravno programsko rješenje. Kvadratna jednadžba gradivo je matematike koje se uči u srednjoj školi, ali svoju široku primjenu ima tijekom cijelog života, osobito u struci inženjera i znanstvenika. S obzirom na potrebe brzog dolaska do rješenja kvadratne jednadžbe, postoje mnoge aplikacije za računalo i mobitel te kalkulatori s ugrađenim funkcijama traženja rješenja kvadratne jednadžbe. Stoga izrada ovakve aplikacije ima prvenstveno u cilju unaprijediti znanje matematike i programiranja te snalaženja prilikom izrade gotovog programskog rješenja s grafičkim sučeljem. Izrada ove aplikacije ne podrazumijeva istraživanje novih područja, već primjenu i proširivanje znanja stečenog tijekom studija u svrhu izrade konkretnog i funkcionalnog programa.

Programski dio izrađen je u razvojnom okruženju Visual Studio koji se smatra jednim od najmoćnijih okruženja za izradu aplikacija na Windows operacijskom sustavu. Upotrijebljen je programski jezik Visual C# koji ima mogućnost povezivanja s Windows formama. Na taj se način omogućila relativno jednostavna izrada grafičkog sučelja aplikacije.

Zadatak rada je bio za unesenu kvadratnu jednadžbu ispisati njezine nul-točke, diskriminantu i koordinate tjemena te iscrtati graf zadane funkcije. Potrebno je omogućiti unos i druge kvadratne jednadžbe te ispisati koordinate sjecišta i međusobni odnos dviju zadanih parabola. Obje parabole potrebno je prikazati u koordinatnom sustavu. Nakon testiranja aplikacije sa zadanim različitim koeficijentima zaključuje se kako aplikacija ispravno radi. Svi se parametri računaju točno i precizno, a graf funkcije je pametno smješten u koordinatni sustav, tako da je os simetrije u središtu prikazanog dijela koordinatnog sustava. Kada su ucrtane dvije parabole, obje su vidljive u koordinatnom sustavu.

Postoje određene mogućnosti unapređenja izrađene aplikacije. Moguće je vizualno obogatiti grafičko sučelje. Ideja je, također, unaprijediti graf kako bi se klikom miša po njemu moglo kliziti te očitavati koordinate točke na određenom položaju kursora miša.

LITERATURA

[1] Gusić Jelena, Mladinić Petar, Pavković Boris, **MATEMATIKA 2**, Školska knjiga, Zagreb, 2006.

[2] How Two Parabolas Intersect, www.uzinggo.com/how-two-parabolas-intersect/graphing-quadratic-functions/algebra-1

[3] Visual Studio, www.visualstudio.com

SAŽETAK

Kvadratna jednadžba svoju široku primjenu ima u inženjerstvu i znanostima srodnim matematiци. U radu se prikazuje teorijska matematička podloga kvadratne jednadžbe i kvadratne funkcije te njezina primjena. Izrađena je desktop aplikacija u programskom jeziku C# unutar razvojnog okruženja Visual Studio koja za zadanu kvadratnu jednadžbu prikazuje njezin graf te očitava nultočke, diskriminantu i tjeme. Na ovaj se način brzo dolazi do navedenih parametara zadane kvadratne jednadžbe. Za zadane dvije kvadratne jednadžbe računaju se koordinate njihova presjeka, ukoliko ono postoji.

Ključne riječi:

kvadratna jednadžba, Visual C#

ABSTRACT

Title:

Application for quadratic equation

Quadratic equation is widely applicable in engineering and science. This thesis interprets theoretical background of quadratic equation and quadratic function, as well as its usage. A desktop application is created in development environment Visual Studio, in programming language C#. Application takes given equation and shows its graph, computes its roots, discriminant and vertex. This speeds up process of getting wanted parameters. If there are two equations given, intersection is computed, if parabolas intersect.

Keywords:

quadratic equation, Visual C#

ŽIVOTOPIS

Martin Juzbašić rođen je 15. travnja 1988. godine u Slavonskom Brodu. U Županji je živio do 2007. godine gdje je završio OŠ Ivan Kozarac i Tehničku školu, smjer Strojstvo. Iste godine upisuje Stručni studij elektrotehnike, smjer Informatika, na Elektrotehničkom fakultetu u Osijeku. Tijekom studija aktivno sudjeluje na brojnim sportskim natjecanjima u rukometu predstavljajući fakultet i sveučilište te pritom osvaja brojne medalje i nagrade, a najistaknutije je drugo mjesto na 1. Europskoj Univerzijadi u Cordobi 2012.godine. Dvije godine za redom (2011. i 2012.) proglašavan je najuspješnijim sportašem Sveučilišta J.J.Strossmayer, a 2012. godine Hrvatski akademski sportski savez dodjeljuje mu godišnju nagradu za izniman doprinos promociji i razvoju akademskog sporta.

PRILOZI

Relevantan dio koda u kojem se nalazi implementacija prikaza kvadratnih jednadžbi te izračuna njihovih parametara:

```
//klik na lijevi OK izvrsava sljedeci kod:
private void button1_Click(object sender, EventArgs e)
{
    double a,b,c;
    //provjera jesu li uneseni realni brojevi:
    bool isDoublea = double.TryParse(textBox1.Text, out a);
    if (isDoublea && a!=0)
    {
        bool isDoubleb = double.TryParse(textBox2.Text, out b);
        if(isDoubleb)
        {
            bool isDoublec = double.TryParse(textBox3.Text, out
                c);
            if(isDoublec)
            {
                //brisanje tocaka (x,y) iz memorije (zbog
                ponovnog pokretanja)
                this.chart1.Series["Jednadzba"].Points.Clear();
                this.chart1.Series["Jednadzba 2"].Points.Clear
                    ();

                //prikaz jednadzbe:
                label13.Text = "Jednadzba: " + a + "x\xB2 +" +
                    b + "x +" + c;

                //rjesenja jednadzbe:
                double x1;
                double x2;

                //diskriminanta
                double D = ((b * b) - (4 * a * c));

                //jedno realno rjesenje
                if (D == 0)
                {
                    x1 = (-b + Math.Sqrt((b * b) - (4 * a * c))
                        ) / (2 * a);
                    label5.Text = "x1 = " + x1.ToString("N4");
                    label6.Text = "x2 = x1";
                    label9.Text = "D = 0 \nPostoji jedna
```



```

        dvostruka \nrealna nultocka.";
    }

    // dva realna rjesenja
    if (D > 0)
    {
        x1 = (-b + Math.Sqrt((b * b) - (4 * a * c))
            ) / (2 * a);
        x2 = (-b - Math.Sqrt((b * b) - (4 * a * c))
            ) / (2 * a);
        label5.Text = "x1 = " + x1.ToString("N4");
        label6.Text = "x2 = " + x2.ToString("N4");
        label9.Text = "D = " + D + " > 0 \nPostoje
            dvije realne nultocke.";
    }

    //dva kompleksna rjesenja
    else if (D < 0)
    {
        double x1_r, x1_i, x2_r, x2_i;
        x1_r = -b / (2 * a); //realni dio rjesenja
        x1_i = Math.Sqrt(-((b * b) - (4 * a * c)))
            / (2 * a); //imaginarni dio rjesenja,
            minus se dodaje da se izbjegne korijen
            iz neg.broja
        x2_r = -b / (2 * a);
        x2_i = -1 * Math.Sqrt(-((b * b) - (4 * a *
            c))) / (2 * a);
        label5.Text = "x1 = " + x1_r.ToString("N2")
            + "+" + x1_i.ToString("N2") + "i";
        label6.Text = "x2 = " + x2_r.ToString("N2")
            + x2_i.ToString("N2") + "i";
        label9.Text = "D = " + D + " < 0 \nPostoje
            dvije konjugirano \nkompleksne nultocke.
            ";
    }

    //koordinate tjemena
    double Tx, Ty;
    Tx = -b / (2 * a);
    Ty = -D / (4 * a);
    label10.Text = "x = " + Tx.ToString("N4");
    label11.Text = "y = " + Ty.ToString("N4");
    if (a > 0) label12.Text = "Tjeme je minimum.";
    else label12.Text = "Tjeme je maksimum.";

```

```

        //prikaz grafa (10 tocaka lijevo i desno od
        tjemena)
        for (int x = (int)Tx - 10; x < (int)Tx + 10; x
            ++)
        {
            double y = (a * x * x) + (b * x) + c; //
            trazi y po formuli
            this.chart1.Series["Jednadzba"].Points.
                AddXY(x, y); //dodaje tocke u graf
            //kasnije ih spaja glatkom linijom (jer je
            tako postavljeno u svojstvima grafa)

        }

        //brisanje koordinata sjecista:
        label25.Text = " ";
        label26.Text = " ";
    }
}

}

//klik na desni OK izvorsava sljedeci kod:
private void button3_Click(object sender, EventArgs e)
{
    //poziva iscrtavanje prve jednadzbe i njezinih parametara
    button1_Click(sender, e);

    //dio koda za drugu jednadzbu:
    double a,b,c;

    bool isDoublea = double.TryParse(textBox4.Text, out a);
    if (isDoublea && a != 0) //a ne smije biti nula
    {
        bool isDoubleb = double.TryParse(textBox5.Text, out b);
        if (isDoubleb)
        {
            bool isDoublec = double.TryParse(textBox6.Text, out
                c);
            if (isDoublec)
            {
                //prikaz jednadzbe:
                label14.Text = "Jednadzba: " + a + "x\xB2 +" +
                    b + "x +" + c;
            }
        }
    }
}

```

```

//rjesenja jednadzbe:
double x1;
double x2;

//diskriminanta
double D = ((b * b) - (4 * a * c));

//jedno realno rjesenje
if (D == 0)
{
    x1 = (-b + Math.Sqrt((b * b) - (4 * a * c))
        ) / (2 * a);
    label19.Text = "x1 = " + x1.ToString("N4");
    label20.Text = "x2 = x1";
    label21.Text = "D = 0 Postoji jedna
        dvostruka realna nultočka.";
}

// dva realna rjesenja
if (D > 0)
{
    x1 = (-b + Math.Sqrt((b * b) - (4 * a * c))
        ) / (2 * a);
    x2 = (-b - Math.Sqrt((b * b) - (4 * a * c))
        ) / (2 * a);
    label19.Text = "x1 = " + x1.ToString("N4");
    label20.Text = "x2 = " + x2.ToString("N4");
    label21.Text = "D = " + D + " > 0 \nPostoje
        dvije realne nultočke.";
}

//dva kompleksna rjesenja
else if (D < 0)
{
    double x1_r, x1_i, x2_r, x2_i;
    x1_r = -b / (2 * a); //realni dio rjesenja
    x1_i = Math.Sqrt(-((b * b) - (4 * a * c)))
        / (2 * a); //imaginarni dio rjesenja,
        minus se dodaje da se izbjegne korijen
        iz neg.broja
    x2_r = -b / (2 * a);
    x2_i = -1 * Math.Sqrt(-((b * b) - (4 * a *
        c))) / (2 * a);
    label19.Text = "x1 = " + x1_r.ToString("N2"
        ) + "+" + x1_i.ToString("N2") + "i";
}

```

```

        label20.Text = "x2 = " + x2_r.ToString("N2"
            ) + x2_i.ToString("N2") + "i";
        label21.Text = "D = " + D + " < 0 \nPostoje
            dvije konjugirano \nkompleksne nultocke
            .";
    }

    //koordinate tjemena
    double Tx, Ty;
    Tx = -b / (2 * a);
    Ty = -D / (4 * a);
    label24.Text = "x = " + Tx.ToString("N4");
    label23.Text = "y = " + Ty.ToString("N4");
    if (a > 0) label22.Text = "Tjeme je minimum.";
    else label22.Text = "Tjeme je maksimum.";

    //prikaz grafa (10 lijevo i desno od tjemena)
    for (int x = (int)Tx - 10; x < (int)Tx + 10; x
        ++)
    {
        double y = (a * x * x) + (b * x) + c; //
            trazi y po formuli
        this.chart1.Series["Jednadzba 2"].Points.
            AddXY(x, y); //dodaje tocke u graf
        //kasnije ih spaja glatkom linijom (jer je
            tako postavljeno u svojstvima grafa)
    }
    }
}
    sjecista(); //poziv funkcije za trazenje sjecista
}

private void sjecista ()
{
    double a,b,c,d,g,f;

    bool isDoublea = double.TryParse(textBox1.Text, out a);
    bool isDoubleb = double.TryParse(textBox2.Text, out b);
    bool isDoublec = double.TryParse(textBox3.Text, out c);
    bool isDoubled = double.TryParse(textBox4.Text, out d);
    bool isDoubleg = double.TryParse(textBox5.Text, out g);
    bool isDoublef = double.TryParse(textBox6.Text, out f);
    //provjera jesu li brojevi realni:
    if (isDoublea && a != 0 && isDoubleb && isDoublec &&
        isDoubled && d != 0 && isDoubleg && isDoublef)

```

```

{

double D = (b - g) * (b - g) - 4 * (a - d) * (c - f);

//ako dobiveni sustav nije kvadratna jedn.:
if(a==d)
{
    //nema rjesenja, mimoilazenje:
    if(b==g && c!=f)
    {
        label25.Text = "Nema presjeka.";
        label26.Text = " ";
        label27.Text = " ";
    }

    //preklapanje:
    else if(b==g && c==f)
    {
        label25.Text = "Parabole su jednake i \
npreklapaju se.";
        label26.Text = " ";
        label27.Text = " ";
    }

    //presjecanje u jednoj tocki:
    else if(b!=g)
    {
        double x1 = (f - c) / (b - g);
        double y1 = a * x1 * x1 + b * x1 + c;

        label25.Text = "x1 =" + x1.ToString("N2") + "\
ny1 =" + y1.ToString("N2");
        label26.Text = " ";
        label27.Text = "Postoji jedno sjeciste.";
    }
}

//ako dobiveni sustav je kvadratna jedn.:
if (a != d)
{
    //presjecanje u dvije tocke, D>0:
    if (D > 0)
    {
        double x1 = (-(b - g) + Math.Sqrt(D)) / (2 * (a
- d));
        double x2 = (-(b - g) - Math.Sqrt(D)) / (2 * (a
- d));
    }
}
}

```

```

double y1 = a * x1 * x1 + b * x1 + c;
double y2 = a * x2 * x2 + b * x2 + c;

label25.Text = "x1 =" + x1.ToString("N2") + "\
ny1 =" + y1.ToString("N2");
label26.Text = "x2 =" + x2.ToString("N2") + "\
ny2 =" + y2.ToString("N2");
label27.Text = "Postoje dva sjecista.";
}

//dodirivanje, D=0
else if (D == 0)
{
double x1 = (-(b - g) + Math.Sqrt(D)) / (2 * (a
- d));
double y1 = a * x1 * x1 + b * x1 + c;

label25.Text = "x1 =" + x1.ToString("N2") + "\
ny1 =" + y1.ToString("N2");
label26.Text = " ";
label27.Text = "Postoji jedno mjesto dodira.";
}

//mimoilazenje, D<0
else if (D < 0)
{
label25.Text = "Nema presjeka.";
label26.Text = " ";
label27.Text = " ";
}
}
}
}

```