Razvoj programskog sustava za proračun vratila

Jagarinec, Roman

Master's thesis / Diplomski rad

2010

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:235:819753

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-28**

Repository / Repozitorij:

Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb

Sveučilište u Zagrebu
Fakultet strojarstva i brodogradnje

DIPLOMSKI RAD

Voditelj rada:
Prof. dr. sc. Neven Pavković

Roman Jagarinec

Zagreb, 2010.

Sveučilište u Zagrebu
Fakultet strojarstva i brodogradnje

DIPLOMSKI RAD

Roman Jagarinec

Zagreb, 2010.

Izjavljujem da sam ovaj rad izradio samostalno služeći se stečenim znanjem i navedenom literaturom.

Posebno se zahvaljujem Prof. dr. sc. Nevenu Pavkoviću na savjetima i vodstvu tokom izrade ovog rada.

Zahvaljujem se poduzeću Promming d.o.o. te direktoru gosp. Mihalju Novinščaku na financijskoj potpori tokom mojeg studija.

Hvala prijateljima na faksu i izvan njega koji su me poticali na rad i bili mi od velike pomoći tokom čitavog studija.

Na kraju najveće hvala čitavoj mojoj obitelji, a posebno roditeljima što su mi omogućili da se školujem i što su pravi uzor u mojem životu.

SAŽETAK

U okviru diplomskog rada prikazana je i objašnjenja implementacija i primjena dvaju različitih programa (CATIA i Microsoft Excel) te programskog jezika Visual Basic sa svrhom skraćivanja vremena proračuna vratila.

Proces skraćivanja vremena proračuna i bržeg razvoja potreban je zbog veće konkurentnosti na tržištu, te izbjegavanja monotonih računanja i ponavljanja istovjetnih radnji, koje bi inače bile dugotrajne za svako vratilo. Na ovaj način moguće je napraviti proračun za čitavu seriju vratila koja se razlikuju u pojedinim dimenzijama te na kraju na temelju iskustva i zahtjeva izabrati najpovoljnije.

Potrebno je napomenuti da programi sami po sebi ne odlučuju koje je vratilo najbolje, odnosno optimalno, već samo daju rezultate proračuna, prema kojima korisnik, odnosno odgovorna osoba odabire željeno vratilo.

Osim vratila program izračunava i vijek trajanja ležaja na osnovu dinamičkih opterećenja koja djeluju na isti.

Tijek rada se sastoji od modeliranja 3D modela u programskom paketu CATIA, gdje se određuje oblik vratila i definiraju parametri. Preko Design Table-a, kroz koji se unose iznosi parametara, a time i dimenzije vratila, povezuje se model sa Visual Basic-om u kojem se vrši obrada ulaznih podataka i sami proračun vratila. U nastavku prikazana i detaljno objašnjena izrada programa te svih sporednih datoteka potrebnih za pravilno funkcioniranje programa.

SADRŽAJ:

1. UVOD 1

2. OPIS KORIŠTENIH PROGRAMA 2

2.1. EXCEL 2
2.2. CATIA 2

3. PARAMETRIZACIJA 3

4. UVOD U PROBLEM 4

4.1. UNOS DIMENZIJA VRATILA PREKO SUČELJA PROGRAMA 6
4.2. UNOS DIMENZIJA VRATILA VRATILAIZRAVNO U „DESIGN TABLE“ 8
4.2.1. RAZLOG TRANSFORMACIJE DIMENZIJA VRATILA 8
4.2.2. OBJAŠNJENJE UNOSA DIMENZIJA IZRAVNO U DESIGN TABLE 9

5. UNOS OSTALIH PODATAKA TE PRIPREMA UNESENIH PODATAKA ZA PRORAČUN PRIJENOSNIM MATRICAMA 19

6. PRIPREMA I OBRADA IZLAZNIH PODATAKA 30

7. ZAKLJUČAK 36

8. PRIMJERI 37

8.1. PRORAČUN SERIJE VRATILA PREDSTAVLJENIM PROGRAMOM 37
8.2. USPOREDBA REZULTATA ANALITIČKI TE PROGRAMOM PRORAČUNATOG VRATILA 44
8.2.1. ANALITIČKO RJEŠENJE 44
8.2.2. PRORAČUN PROGRAMOM U VISUAL BASICU 47

9. LITERATURA 49

10. DODATAK 50

POPIS SLIKA:

Slika 1: *Unos i transformacija dimenzija vratila* 5
Slika 2: *Osnovno sučelje za unos segmenta* 6
Slika 3: *Osnovno sučelje sa unesenim segmentom* 6
Slika 4: *Sučelje za definiranje skošenja* 7
Slika 5: *Sučelje za definiranje radijusa* 7
Slika 6*: Sučelje za definiranje utora za pero* 7
Slika 7: *Sučelje za definiranje utora za uskočnik* 7
Slika 8: : *Model vratila i Design Table u CATIA-i* 8
Slika 9: *Proces unosa dimenzija vratila izravno u Design Table* 10
Slika 10: *Radionički crtež vratila sa slijepim kotama* 12
Slika 11: *Design Table u Microsoft Excelu* 13
Slika 12: *Tablica za odabir serije vratila* 14
Slika 13: *Tablica sa transformiranim dimenzijama* 15
Slika 14: *Tablica sa transformiranim dimenzijama čitave serije vratila* 15
Slika 15: *Shema za transformaciju kota* 16
Slika 16: *Osnovno sučelje glavnog programa* 19
Slika 17: *Vizualni prikaz unesenih dimenzija* 20
Slika 18: *Sučelje za unos dodatne radijalne sile* 21
Slika 19: *Sučelje za unos torzionog momenta* 21
Slika 20: *Open Dialog* 21
Slika 21: *.txt datoteka za ležaj* 24
Slika 22: *Aproksimacija polja* 27
Slika 23: *Unos ostalih podataka te ispis rezultata* 29
Slika 24: *Radni list sa dijagramima* 30
Slika 25: *Radni list sa izlaznim podacima za jedno vratilo* 32
Slika 26: *Shema izrade „dijagram“ stupca* 33
Slika 27: *Brisanje „test“ krivulje* 34
Slika 28: *Dodavanje podataka za novo vratilo* 35
Slika 29: *Odabir naziva i putanje* 35
Slika 30: *Odabir podataka za apscisu* 35
Slika 31: *Konačni izgled dijagrama za 2 vratila* 35
Slika 32: *Podaci o 9. I 19. Vratilu* 38
Slika 33: *Transformirane kote za 9. vratilo* 39
Slika 34: *Transformirane kote za 19. vratilo* 39
Slika 35: *Ulazni podaci* 39
Slika 36: *Dodatno radijalno opterećenje* 40
Slika 37: *Torzioni moment* 40
Slika 38: *Dijagram momenta savijanja i maksimalnog naprezanja na savijanje* 41
Slika 39: *Dijagram poprečne sile i naprezanja na smik* 42
Slika 40: *Dijagram progiba* 43
Slika 41: *Opterećenja na vratilo* 44
Slika 42: *Uklještenje na mjestu 3.* 45
Slika 43: *Uklještenje na mjestu 6.* 46
Slika 44: *Dimenzije analiziranog vratila* 47
Slika 45: *Progib vratila* 48

POPIS TABLICA:

Tablica 1: *Osnovne zadaće cjelokupnog programa* 4
Tablica 2: *Macro procedure*.....16
Tablica 3: *Unos podataka* 19
Tablica 4: *Popis procedura unutar programa* 22
Tablica 5: *Vrste ležaja*.....24
Tablica 6: *Raspodjela segmenata na polja* 25
Tablica 7: *Zarezni faktori*.....26
Tablica 8: *Izlazni podaci* 31

1. UVOD

U današnje vrijeme javlja se potreba čim bržeg plasmana određenog proizvoda na tržište. Smanjeno vrijeme razrade u razvijanja novog proizvoda povećava njegovu konkurentnost na tržištu. Naravno sa smanjenim vremenom razvijanja, automatski se smanjuju i cjelokupni troškovi proizvodnje. Kraj cijelog tog procesa razvijanja potrebno je odabrati optimalno rješenje, koje ne utječe na razinu kvalitete koja mora ostati zadovoljavajuća.

Upravo se iz ovih razloga javila potreba za razvojem raznovrsnih aplikacija koje ubrzavaju određene poslove u postupku konstruiranja. Zadatak ovog diplomskog je razvoj programa koji će automatizirati proračun čvrstoće i progiba serije vratila za elektromotore. Potreba za automatizacijom takve vrste nalazi se u činjenici da su proračuni vratila kod konstruiranja elektromotora vrlo učestali. Naime u nekim slučajevima konstruiranja potrebno je mnogo puta ponoviti proračune s raznim varijantama opterećenja ili različitim izvedbama.

Za proračun se koristilo vratilo koje je modelirano i parametrizirano u programskom paketu CATIA. Microsoft Excel služi kao spona koja povezuje napravljeni model i Visual Basic te na kraju služi za ispis i prikaz rezultata. Obradu ulaznih podataka i sam proračun vratila izvodi program napravljen u Visual Basic-u.

Potrebno je napomenuti da je za korištenje programa nužno barem osnovno poznavanje samog proračuna vratila, a isto tako i rada u CATIA-i te da nije moguće da ga koriste potpuni laici. Isto tako je potrebno napomenuti da je korištenje sučelja napravljenog u Visual Basic-u u mnogočemu pojednostavilo pripremu podataka za proračun. Samo sučelje je oblikovano na korisniku prijazan način i može ga koristiti svaki inženjer. Osim analize vratila proračun izračunava i vijek trajanja odabranih ležaja.

Svi rezultati koji su dobiveni za vratila prikazani su ispisom u Microsoft Excel-u. U istoj .xls datoteci nalazi se usporedba naprezanja serije vratila i to u obliku zajedničkih dijagrama te je na taj način moguće izabrati optimalno.

Moguće je primijetiti da se na ovaj način uvelike smanjuje vrijeme potrebno za proračun serije vratila što nam je i bio cilj. U daljnjem tijeku rada biti će predstavljen postupak izrade gore predstavljenog programa, kao i koraci koje program obavlja tokom izvedbe.

2. OPIS KORIŠTENIH PROGRAMA

2.1.EXCEL

Microsoft Excel uglavnom služi za rješavanje problema matematičkog tipa pomoću tablica i polja koje je moguće povezivati različitim formulama. Može poslužiti i za izradu jednostavnijih baza podataka (program koji služi za izradu složenijih baza podataka je Microsoft Access). Na temelju unesenih podataka, lako iz tablica može stvarati grafikone. Također omogućuje dodavanje različitih objekata: tablica, slika, grafikona... Česta mu je primjena u uredima, gdje služi za izradu troškovnika,obračuna i sl. Excel radni list se sastoji od redaka i stupaca. Sjecište redaka i stupaca nazivaju se ćelije. lit.[1]

Budući da je Excel kao sastavni dio Microsoft Office-a, jedan od osnovnih alata za obradu tekstualnih podataka te je poznavanje istog jedna od osnova opće informatičke pismenosti, nema ga potrebe dodatno opisivati.

2.2.CATIA

U svijetu industrijskih dizajnera naći ćete samo tri važeća programska paketa koja zauzimaju ogroman tržišni udio. Radi se, dakako, o poznatom i nadaleko i naširoko hvaljenom i-DEAS-u, ništa slabijem ProEngineer-u. CATIA je danas etablirani programski paket koji je svoju primjenu ponajprije našao u transportnoj industriji, što i nije toliko čudno s obzirom na činjenicu da tvorac CATIA-e jedan od najvećih europskih proizvođača aviona.

Riječ je, dakle, o francuskom Dassaultu, koji je s razvojem CATIA-e počeo još sedamdesetih godina. Tada je cilj bio stvaranje programa za 3D modeliranje i proizvodnju prvenstveno za internu upotrebu unutar tadašnjeg Dassault Aviationa. U svrhu daljnjeg razvoja CATIA-e osnovana je posebna tvrtka Dassault Systemes, koja je nakon nekog vremena ušla i u sastav velikog IBM-a, pod čijim se okriljem danas nastavlja razvoj jednog od najuspješnijih paketa za 3D modeliranje. Prodaja i podrška korisnicima CATIA-e odvija se upravo preko IBM-ove mreže poslovnica ili poslovnih partnera koji su specijalizirani za pružanje tehničke potpore. No, trebalo je otprilike petnaestak godina da CATIA nađe svoju primjenu i izvan zrakoplovne industrije. CATIA-u, po posljednjim podacima, koristi čak sedamnaest od 25 najvećih svjetskih proizvođača automobila. Na primjer, europska grana DaimlerChryslera ima instaliranih više od dvije i pol tisuće CATIA radnih stanica; čitava Volkswagen grupa, BMW, Renault, PSA grupa i Fiat Auto koriste CATIA-u kao svoj osnovni CAD/CAM alat. Koliko je CATIA moćan paket govori i činjenica da je koriste Japanci, poznati po napretku tehnologije, čiji se najveći proizvođači Toyota i Mitsubishi ponajviše oslanjaju upravo na CATIA-u. U tu veliku grupu spadaju i Ford Motor Company te Hyundai, dakle još dva od vodećih pet svjetskih proizvođača automobila. Logično, iz industrijskih će se hala CATIA-ine mogućnosti prenijeti i na trkaće staze i to - ni više ni manje nego - u najpopularniji automobilski sport, Formulu 1. Za konstruiranje šasija svojih bolida koristi je šest od ukupno jedanaest konstruktora, a za konstrukciju motora čak šest od ukupno osam dobavljača. Švedska Scania ima pak 350 instaliranih CATIA radnih stanica koje, naravno, koristi za razvoj svojih teških kamiona čija je brojka dosegla čak jedan milijun. U CATIA-i je razvijen zaštitni znak tih švedskih kamiona, V8 motor. čak su i tradicionalno konzervativni proizvođači kao što su Rolls Royce ili Bentley pribjegli upotrebi CATIA-e, premda se cijela njihova filozofija pravljenja automobila zasniva na ručnom sklapanju. lit. [2]

3. PARAMETRIZACIJA

Parametrizacija je, kao pomoć pri ubrzavanju i pojednostavljenju dobivanja dokumentacije za određeni proizvod, postala aktualna u zadnjih desetak godina sa razvojem odgovarajućih programskih aplikacija. Nakon što su provedena određena istraživanja lit. [3] na temu koji dio razvoja proizvoda oduzima najviše vremena inženjera, zaključeno je da otprilike 80% vremena inženjeri provode prepravljajući postojeće modele. Samo ostalih 20% vremena troši se na oblikovanje novih proizvoda, s time da u određenim slučajevima ta brojka može pasti i na 0%. Ta činjenica nam daje do znanja da je upravo to područje inženjerskog rada potrebno što više automatizirati. Upravo tu dolazi do izražaja sama parametrizacija. Prema istraživanju lit. [4] provedenom u tvrtci „Hyundai-Kia“ za konstrukcijski odsjek, parametrizacijom je uštađeno preko tri milijuna dolara godišnje, što je razmjerno 51 000 radnih sati inženjera. Mora se napomenuti da je ovakva ušteda bila omogućena uz dodatno uspostavljanje zajedničke baze podataka za sve inženjere, kako bi svi mogli koristiti već postojeće proizvode. Osima što se parametrizacijom proizvoda dobiva na brzini, dobiva se i na poboljšanju kvalitete proizvodnje. Naime neke postavke, kao što su vrsta materijala i primjerice raspored rupa, nije potrebno birati svaki puta iznova. Također se mogu definirati određena pravila koja ograničuju model u smislu da program ne dozvoli nepravilno oblikovanje. Točnije da rupe ne budu preblizu rubovima i slično.

Kod upravljanja parametrima te dodavanja novih veličina za određene parametre najčešće se koristi CATIA-in alat „Design Table“. Kako bi čitaocu bile jasnije sve prednosti Design Table-a, one su navedene u nekoliko sljedećih natuknica.

Design table:

- *Omogućuje vam stvaranje i upravljanje familijom proizvoda. Takve familije mogu primjerice biti proizvodi istog tipa, a da se razlikuju u dimenzijama.*
- *Alat koji prvenstveno služi za olakšavanje pri definiciji konstrukcijskih dijelova. Dostupan je svim korisnicima programskog paketa CATIA, no najupotrebljiviji u aplikaciji KNOWLEDGE ADVISOR. Design table se može napraviti koristeći CATIA datoteke, u tom slučaju se govori o ispisu podataka u Design table. Također se može primijeniti na dokumentu tj. podaci su uneseni u CATIA dokument iz Design table-a.*
- *Koristi vrijednosti iz vanjskih dokumenta te ih pridodaje parametrima u CATIA-i. Vanjski dokument može biti spremljen kao TXT ili kao EXCEL datoteka. Vrlo je važno, u korištenju Design table-a, pravilno povezati parametre u CATIA datoteci sa veličinama u tablici. Neke veličine u eksternoj datoteci možda nisu kompatibilne s parametrima u CATIA-i. U tom slučaju se samo dio vanjskih vrijednosti povezuje sa parametrima. Veze se stvaraju deklariranjem takozvanih asocijacija (veza).*
- *Potencijali Design table-a se u potpunosti mogu iskoristi tek u sklopu Knowledge Advisor aplikacije. Korisniku su u tom slučaju dane funkcije za stvaranje i čitanje parametara. Iste se mogu koristiti u programiranju provjera i pravila. Korištenjem tih funkcija ujedno izbjegavamo potrebu za stvaranjem asocijacija.*

4. UVOD U PROBLEM

Osnovni zadatak ovog rada je priprema svih ulaznih podataka za proračun vratila prijenosnim matricama te izrada potrebnih dijagrama na temelju izlaznih podataka. Ulazni podaci mogu se podijeliti u više kategorija budući da njihov izvor nije isti. Kategorije unosa ulaznih podataka te sve ostale zadaće ovog rada prikazane su tablicom [1].

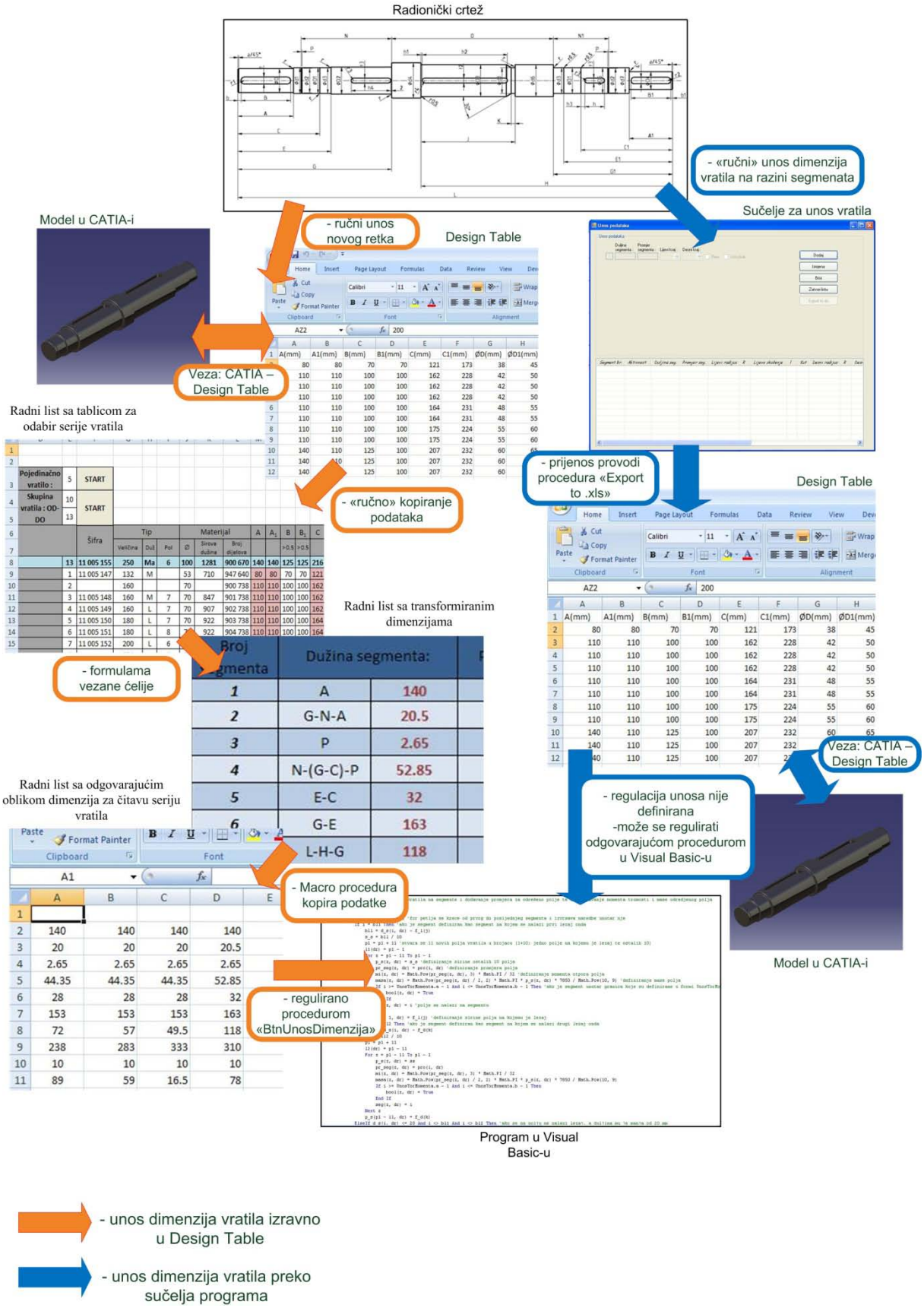
OSNOVNE ZADAĆE CJELOKUPNOG PROGRAMA
- spajanje parametriziranog modela vratila, u programskom paketu CATIA, sa proračunom vratila pomoću prijenosnih matrica
- prilagodba kota gore navedenog vratila te njihov unos u program za proračun metodom prijenosnih matricama
- unos dimenzija te ostalih karakteristika ležaja potrebnih za uležištenje vratila
- unos ostalih podataka potrebnih za analizu vratila (materijal vratila, opterećenja, brzina vrtnje i dr.)
- podjela segmenata vratila na polja/diskove kao elementarne dijelove za proračun prijenosnim matricama
- proračun vratila metodom prijenosnih matrica
- automatsko biranje odgovarajuće širine ležaja s obzirom na ekvivalentno opterećenje ležaja
- automatizirati proračun serije vratila te izrada preglednog prikaza rezultata analize
- izrada zajedničkih dijagrama za čitavu seriju vratila na temelju rezultata dobivenih provedenom analizom

Tablica 1: ***Osnovne zadaće cjelokupnog programa***

Krenuvši od problema unosa dimenzija vratila tj. transformacije i obrade podataka iz CATIA-e u oblik pogodan za proračun prijenosnim matricama, razvijena su dva rješenja unosa dimenzija vratila.

Prvo je rješenje napravljeno na pretpostavki da korisnik nije dovoljno upoznat sa programskim paketom CATIA i ne zna napraviti model vratila te ga pravilno parametrizirati. U tu svrhu je u CATIA-i napravljen model vratila od dvadeset segmenata sa svim potrebnim parametrima. Model vratila može se shvatiti kao „Prazan model“ tj. model čije dimenzije još nisu definirane. Iznos samih dimenzija te aktivnosti određenih značajki na vratilu su povezani preko Design table-a sa programom napravljenim u Visual Basic-u. Preko sučelja prethodno spomenutog programa unose se svi potrebni podaci. Sučelje je oblikovano na način da korisniku nije potrebno gotovo nikakvo znanje rada u CATIA-i. Naime svi potezi koje je potrebno napraviti u CATIA-i objašnjeni su u nastavku ovog rada. U nastavku ovaj će se oblik unosa nazivati „Unos dimenzija vratila preko sučelja programa“.

U drugom načinu unosa podataka također se kreće od radioničkog crteža vratila, na temelju kojeg se modelira te parametrizira vratilo u CATIA-i. Dimenzije vratila koje očitavamo s tehničkog crteža unose se izravno u Design Table, čime definiramo parametre modela. U Excel radnoj knjizi koja sadrži Design Table dimenzije se transformiraju u oblik pogodan za proračun prijenosnim matricama te se takve učitavaju u program. Ovaj način unosa je, prema dogovoru sa mentorom, odabran kao osnovi tj. primarni način i u potpunosti je usklađen sa programom za proračun, a naziva se „Unos dimenzija izravno u Design Table“. Vizualni prikaz oba unosa dimenzija vratila predočen je slikom [1].

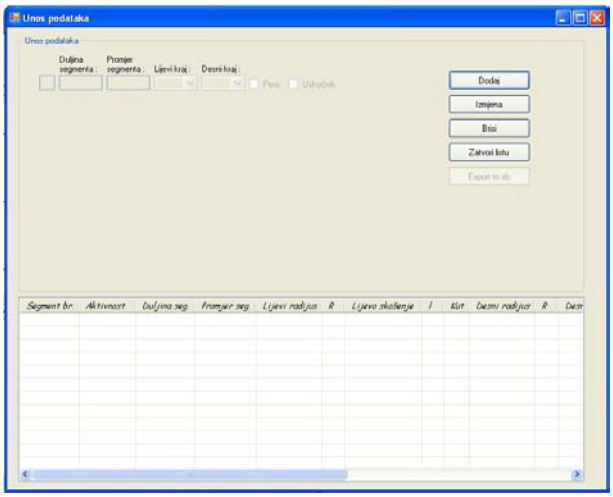


Slika 1: ***Unos i transformacija dimenzija vratila***

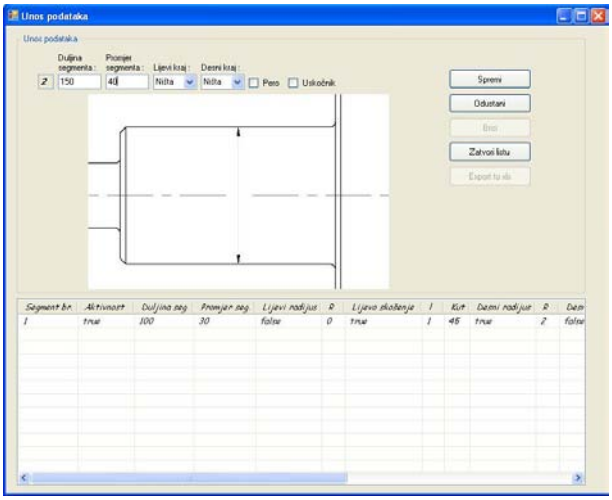
U nastavku rada prikazan je i opisan unos podataka u program koristeći oba prethodno spomenuta rješenja. Za svako od pojedinih rješenja biti će objašnjen način korištenja sučelja te će se na taj način korisnik upoznati i sa svim procedurama sučelja.

4.1. UNOS DIMENZIJA VRATILA PREKO SUČELJA PROGRAMA

Na slici [2] je prikazano osnovno sučelje za unos dimenzija vratila. Unos je koncipiran tako da se unose podaci za svaki segment posebno. Time smo omogućili izradu modela koji nije ovisan o broju segmenata ili o položaju pera te uskočnika na vratilu. Čitava funkcija i sve pogodnosti ovog unosa najbolje će biti objašnjene na način da se prikaže postupak unosa jednog vratila. Važno je još napomenuti da se na ovaj način mogu unositi podaci, a time i dobivati modeli za vratila čiji je broj segmenata manji od 21. Razlog tome nalazi se u činjenici da gotov model vratila u CATIA-i, kojega mi oblikujemo podacima iz Design Table-a, ima 20 segmenata. Ovaj broj nije konačan i vratilo se može povećati, ali za potrebe diplomskog je 20 segmenata zadovoljavajuća brojka.



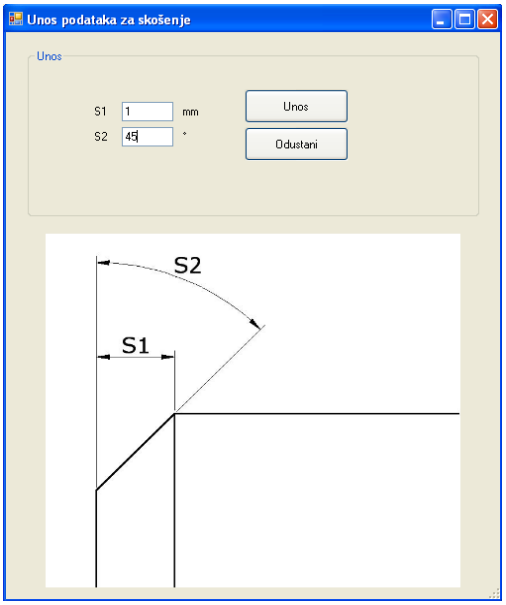
Slika 2: ***Osnovno sučelje za unos segmenta***



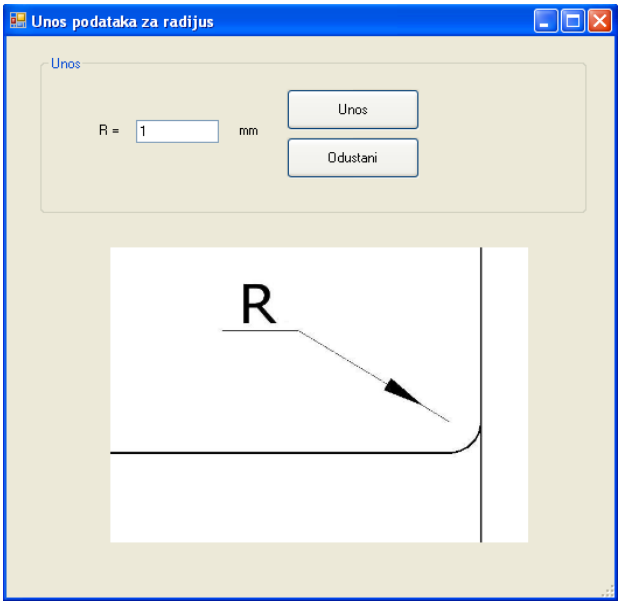
Slika 3: ***Osnovno sučelje sa unesenim segmentom***

Dakle cjelokupni unos podataka kreće od osnovnog sučelja. Na desnoj strani sučelja vidljive su opcije pomoću kojih upravljamo sučeljem. Samo upravljanje se sastoji od opcija pomoću kojih dodajemo, mijenjamo, odnosno brišemo pojedine segmente vratila. Pritiskom na opciju „Dodaj“ aktiviraju se potrebna polja za unos teksta koja služe za unos dimenzija pojedinih segmenata, što je prikazano slikom [3]. Prilikom radnje dodavanja, umjesto opcije „Dodaj“ pojavljuje se nova opcija „Spremi“, pomoću koje se memorira definirani segment. U sljedećem koraku se redom unose dimenzije segmenta. Radi lakše preglednosti i korištenja sučelja, u samom sučelju nalaze se slike koje korisnika referiraju na dimenziju koju unosi. Također postoji lista u kojoj su prikazani svi memorirani segmenti te se oni mogu kasnije provjeravati, mijenjati i eventualno brisati.

Biranjem ruba segmenta u ComboBox opcijama „Lijevi kraj“, odnosno „Desni kraj“ otvara se potpuno novo sučelje, pomoću kojih se određuje da li na tom mjestu postoji skošenje, odnosno radijus te kolika je vrijednost istih. Prikaz novih sučelja te unos podataka prikazani su slikama [4] i [5].

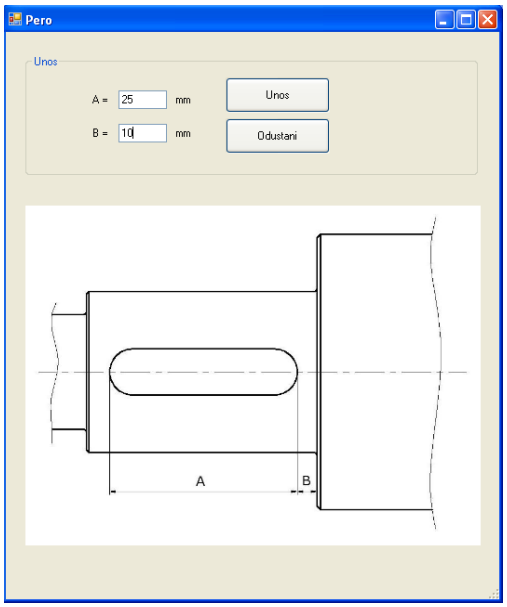


Slika 4: ***Sučelje za definiranje skošenja***

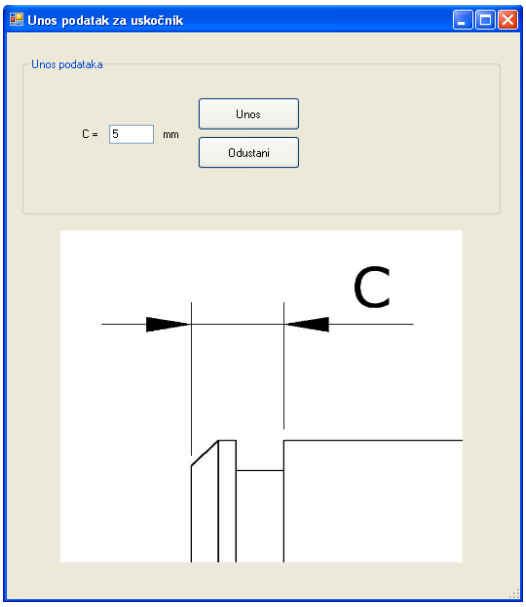


Slika 5: ***Sučelje za definiranje radijusa***

Također postoje opcije pomoću koji je moguće odabrati da li se na određenom segmentu nalazi pero, odnosno uskočnik. Prilikom odabira tih opcija, također se pojavljuju nova sučelja preko kojih se unose potrebne dimenzije, što je prikazano slikama [6] i [7].



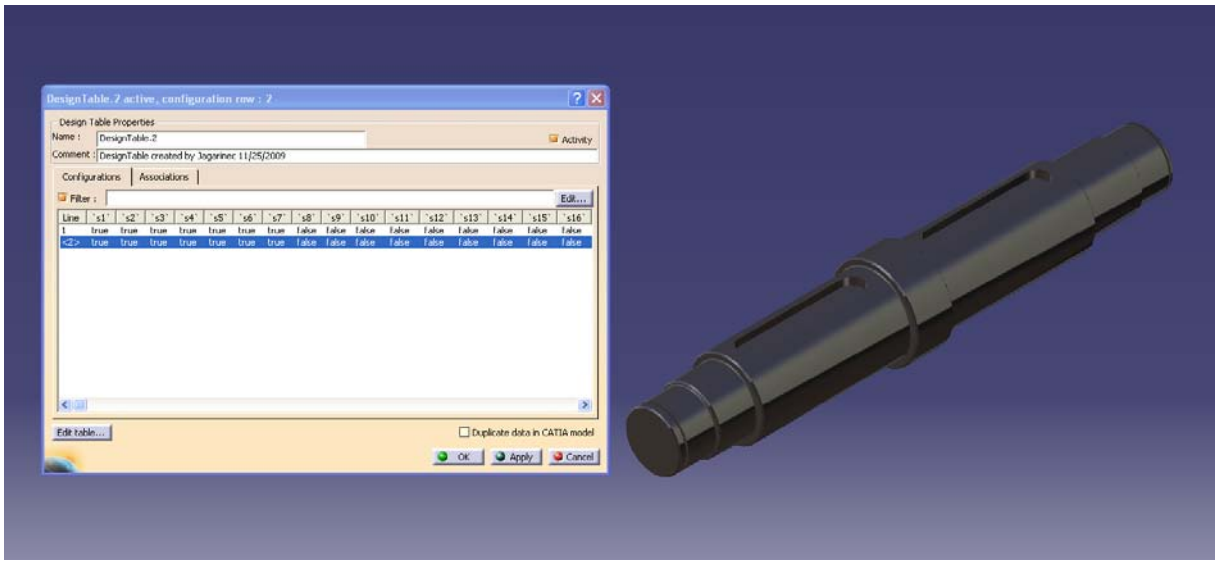
Slika 6*: Sučelje za definiranje utora za pero*



Slika 7: ***Sučelje za definiranje utora za uskočnik***

Naravno sve ove opcije se također unose u prije spomenutu listu i moguće ih je tamo sve provjeriti, i na isti način je moguće upravljanje sa pojedinim segmentima. Sve ostale dimenzije tj. dimenzije koje nisu kotirane na slikama, a potrebne su za definiranje gore prikazanih utora, dobivaju se iz gotovih tablica, koje su izvađene iz tehničke literature [5], a ovise o promjeru segmenta. Nakon spremanja svakog segmenta naredbe se vračaju u početni položaj i moguć je unos novog i izmjena ili brisanje postojećeg segmenta. Kada korisnik unese sve segmente određenog vratila, primoran je zatvoriti listu prije nego je pošalje u Design Table. Zatvaranjem liste automatski se definiraju parametri ostalih segmenata tj. segmenata koji su deaktivirani u modelu. Drugim riječima, ako imamo vratilo od 15 segmenata, posljednjih 5 segmenata na vratilu u CATIA-i mora biti deaktivirano kako bi dobili odgovarajući model vratila.

Samim zatvaranjem liste onemogućava se biranje opcija „Dodaj“, „Izmjena“ i „Briši“, a istovremeno se omogućuje odabir opcije „Ispis u .xls“, kojom se podaci o vratilu unose u željeni Design Table. Opcija „Zatvori listu“ je povratna i lista se može naknadno otvoriti biranjem iste tipke koja je promijenila ime u „Otvori listu“. Pritiskom na opciju „Export to .xls“ otvara se „Open Dialog“ kojim biramo .xls datoteku tj. Design Table u koji se spremaju uneseni podaci. Razlike između aktivirane i deaktivirane, odnosno razlika između opcije čije je biranje omogućeno i onih čije nije, vidljive su iz slika [2] i [3]. Na slici [3] jasno se vidi kako je opcija „Export to .xls“ bljeđa od ostalih što znači da ju nije moguće odabrati. Na kraju nam preostaje samo otvoriti model vratila u CATIA-i i aktivirati željenu verziju, što je prikazano slikom [8].



Slika 8: : *Model vratila i Design Table u CATIA-i*

Daljnja obrada podataka za ovaj način unosa nije obrađena, budući da je u dogovoru sa mentorom drugi način odabran kao bolji te je on obrađen detaljno. Kako bi zaključili poglavlje o ovoj verziji unosa nabrojimo još jednom sve pogodnosti iste. Dakle ovaj program nam omogućuje da bez gotovo ikakvog predznanja rada u programskom paketu CATIA napravimo model vratila. Također kod ovog načina unosa podataka samo vratilo je unaprijed raščlanjeno na razini segmenata, te kod takvog unosa podataka nije potrebno u kasnijoj fazi rada transformirati dimenzije u pogodan oblik. Taj način unosa sadrži u sebi nedostatak koji se manifestira time da je potrebno na papiru raščlaniti vratilo na potrebne segmente iz radioničkog crteža te ih potom unositi kako je prethodno opisano.

Upravo je prije navedeni nedostatak jedan od razloga zbog kojih je drugi način unosa podataka odabran kao odgovarajući te je opisan u nastavku ovog rada.

4.2. UNOS DIMENZIJA VRATILA VRATILAIZRAVNO U „DESIGN TABLE“

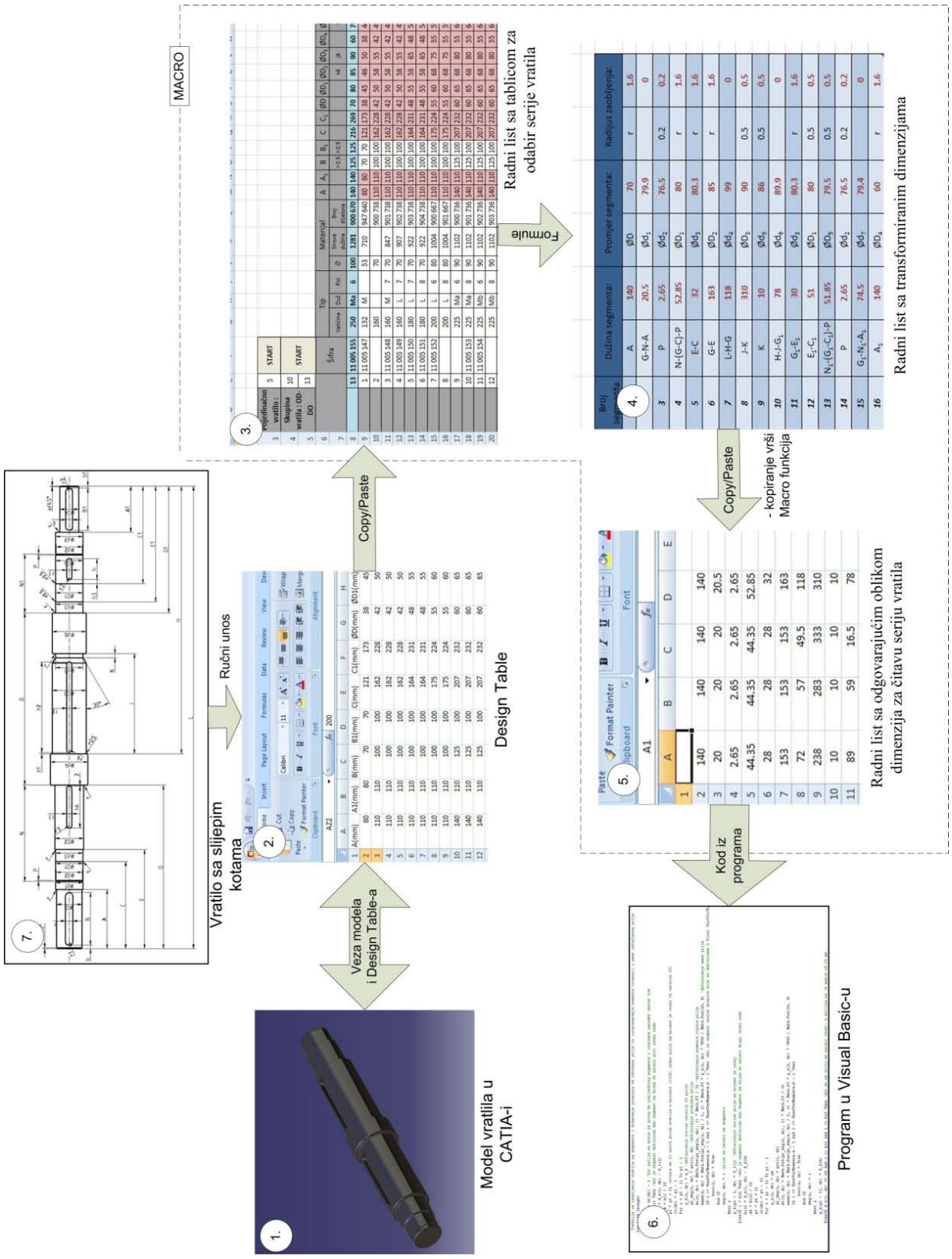
4.2.1. RAZLOG TRANSFORMACIJE DIMENZIJA VRATILA

Ranije smo kao jednu od prednosti „Unosa podataka kroz sučelje programa za unos“ naveli unos podataka na razini segmenata. Time se misli na činjenicu da se u tom obliku unosa, duljine, promjeri i svi ostali podaci koji opisuju segment, definiraju zasebno i neovisno o susjednim segmentima. Napomenuli smo i nedostatak istog u činjenici da je tokom unosa konstruktor primoran, na papiru, transformirati kote iz tehnički ispravnog oblika u oblik gdje one definiraju svaki segment zasebno. U slučaju potrebe za analizom serije vratila takav bi način unosa bio vrlo dugotrajan upravo zbog mučnog posla transformiranja dimenzija.

Koliko god transformacija kota u gore opisan oblik bila mučna, ona je istovremeno nužna, ne samo zbog modela vratila u CATIA-i, nego i zbog programa za proračun vratila metodom prijenosnih matrica. Opis svakog segmenta zasebno omogućava nam kasniju podjelu segmenata na polja što je i osnovna zamisao kod proračuna metodom prijenosnih matrica. Kako bi se izbjeglo ručno transformiranje dimenzija, a time ubrzao unos podataka koji definiraju vratilo, razvijen je „Unos dimenzija vratila izravno u Design Table“.

4.2.2. OBJAŠNJENJE UNOSA DIMENZIJA IZRAVNO U DESIGN TABLE

Čitav postupak, odnosno koraci koji su izvedeni u ovom načinu unosa, prikazani su slikom [9].

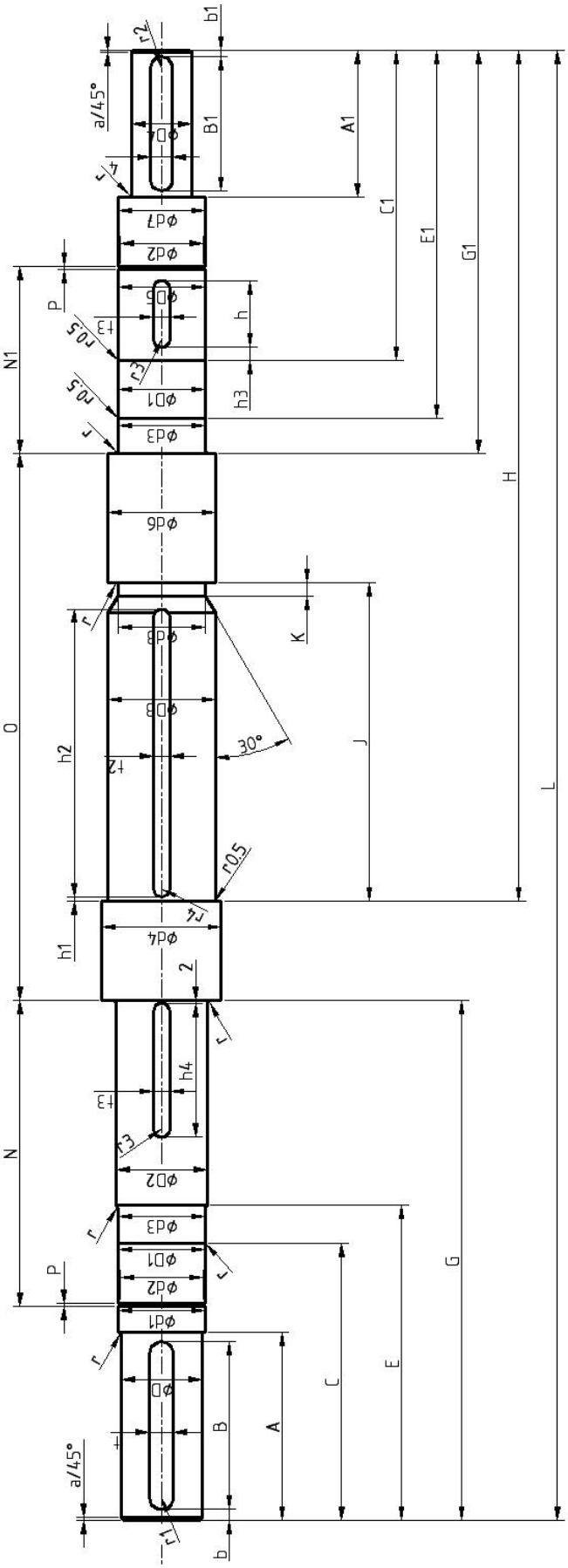


Slika 9: ***Proces unosa dimenzija vratila izravno u Design Table***

Kako bi lakše objasnili interakcije između komponenata dijagrama tj. interakcije između modela vratila, radnih listova te programa za proračun vratila, a samim time i „Unos dimenzija vratila izravno u Design Table“, moramo prvo definirati iste.

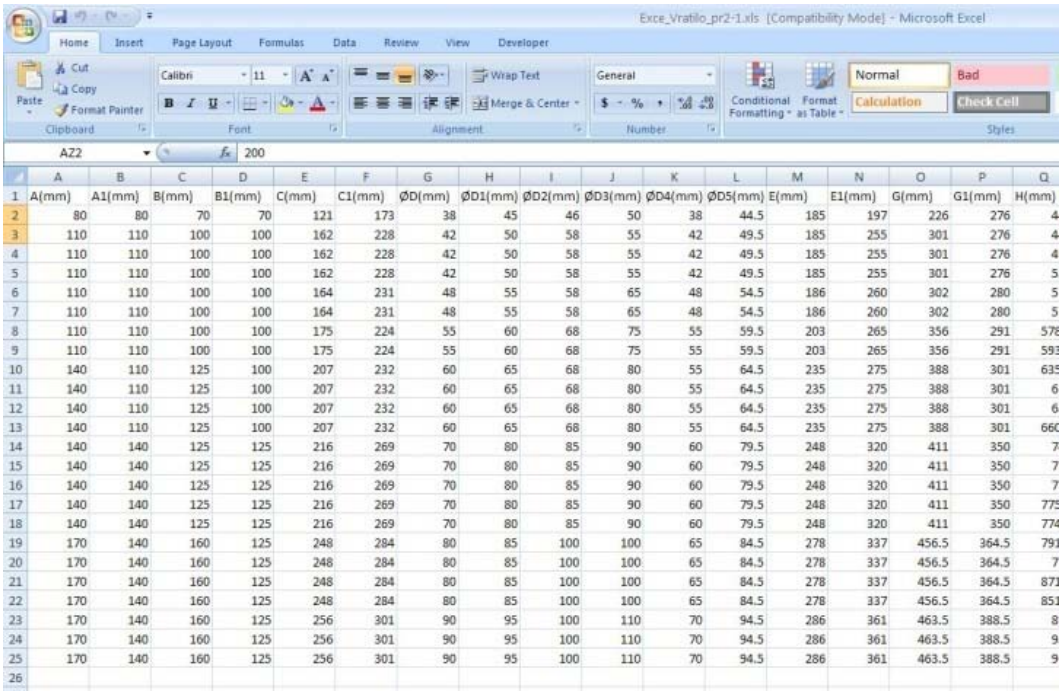
Komponente sustava za modeliranje vratila i transformaciju kota redoslijedom obilježenim na slici [9]:

1. CATIA model [8] – parametrizirani model vratila
2. Design Table [11] – Excel radni list u kojemu su definirani parametri vratila u CATIA-i. Design Table ne smije sadržavati nikakve druge forme (tablice, Macro opcije i sl.) ili podatke, osim imena parametara te njihovih vrijednosti.
3. Radni list sa tablicom za odabir serije vratila [12]– Sadrži identične podatke o vratilu kao i Design Table, no potreban je iz gore navedenog razloga koji sprječava većinu izmjena na Design Table-u. Ovaj radni list nam omogućava vizualnu izmjenu tablice koja sadrži dimenzija vratila te dodavanje „Button-a“ za pokretanje Macro opcija.
4. Radni list sa transformiranim dimenzijama (kotama) [13]– Sadrži tablicu čije su ćelije formulama povezane sa tablicom i Radnog lista za odabir dimenzija. Tablica sadrži dimenzije vratila u obliku pogodnom za proračun metodom prijenosnih matrica.
5. Radni list sa odgovarajućim oblikom dimenzija za čitavu seriju vratila [14] – Služi za spremanje novog opisa vratila nakon njegove transformacije, što znači da sadrži dimenzije segmenata za odabranu seriju vratila te je stoga ulazna datoteka za program koji proračunava vratilo.
6. Program za proračun vratila – Program koristi gore opisani radni list (5.) kao ulaznu datoteku. Odabir i unos podataka iz .xls datoteke koja sadrži gore opisane radne listove pokreće se preko sučelja programa.
7. Radionički crtež [10] – radionički crtež sa slijepim kotama



Slika 10: ***Radionički crtež vratila sa slijepim kotama***

Na slici [9] je vidljivo kako se dimenzije vratila iz radioničkog crteža (7.) unose izravno u Design Table (2.) te su zato model vratila u CATIA-i i njegovi parametri definirani sukladno tome. Modeliranje i parametrizaciju određenog vratila potrebno je samostalno napraviti pa se pretpostavlja da korisnik ima određeno znanje rada u CATIA-i. Vezu između modela vratila i Design Table-a kao prvog radnog lista u .xls datoteci stvara CATIA i nije je potrebno posebno objašnjavati. Dovoljno je reći da svaki stupac u Design Table-u sadrži podatke za jednu kotu iz radioničkog crteža, što znači da definira jedan parametar u modelu. Prijenos podataka iz Design Table-a u „Radni list sa tablicom za odabir serije vratila“ (3.) obavlja se „ručno“ i to tako da se jednostavno odabere redak ili više njih iz Design Table-a te se kopira u slobodni redak u tablici. Takav prijenos je moguć zato što obje tablice imaju isti raspored stupaca koji sadrže vrijednosti dimenzija. U tablici za odabir serije vratila nalaze se dodatne informacije o vratilu poput rednih i serijskih brojeva vratila i slično. Kako te informacije nismo unijeli u Design Table-u treba ih unijeti naknadno nakon kopiranja redaka sa dimenzijama. Time su uneseni svi potrebni podaci o vratilu koje možemo očitati iz tehničkog crteža. Jasno je da je takav unos daleko brži nego „Unos podataka kroz sučelje programa za unos“, a posebno je koristan kod unosa serije vratila.

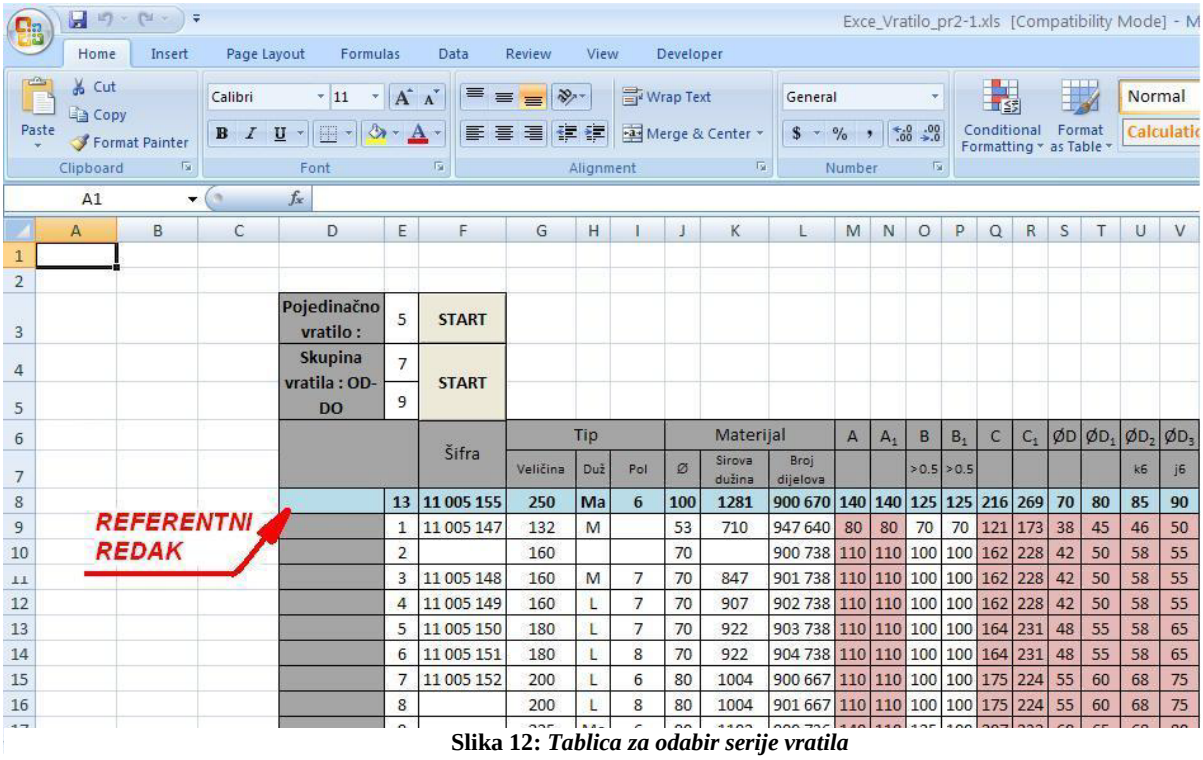


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	A(mm)	A1(mm)	B(mm)	B1(mm)	C(mm)	C1(mm)	ØD(mm)	ØD1(mm)	ØD2(mm)	ØD3(mm)	ØD4(mm)	ØD5(mm)	E(mm)	E1(mm)	G(mm)	G1(mm)	H(mm)
2	80	80	70	70	121	173	38	45	46	50	38	44.5	185	197	226	276	4
3	110	110	100	100	162	228	42	50	58	55	42	49.5	185	255	301	276	4
4	110	110	100	100	162	228	42	50	58	55	42	49.5	185	255	301	276	4
5	110	110	100	100	162	228	42	50	58	55	42	49.5	185	255	301	276	5
6	110	110	100	100	164	231	48	55	58	65	48	54.5	186	260	302	280	5
7	110	110	100	100	164	231	48	55	58	65	48	54.5	186	260	302	280	5
8	110	110	100	100	175	224	55	60	68	75	55	59.5	203	265	356	291	578
9	110	110	100	100	175	224	55	60	68	75	55	59.5	203	265	356	291	593
10	140	110	125	100	207	232	60	65	68	80	55	64.5	235	275	388	301	635
11	140	110	125	100	207	232	60	65	68	80	55	64.5	235	275	388	301	6
12	140	110	125	100	207	232	60	65	68	80	55	64.5	235	275	388	301	6
13	140	110	125	100	207	232	60	65	68	80	55	64.5	235	275	388	301	660
14	140	140	125	125	216	269	70	80	85	90	60	79.5	248	320	411	350	7
15	140	140	125	125	216	269	70	80	85	90	60	79.5	248	320	411	350	7
16	140	140	125	125	216	269	70	80	85	90	60	79.5	248	320	411	350	7
17	140	140	125	125	216	269	70	80	85	90	60	79.5	248	320	411	350	775
18	140	140	125	125	216	269	70	80	85	90	60	79.5	248	320	411	350	774
19	170	140	160	125	248	284	80	85	100	100	65	84.5	278	337	456.5	364.5	791
20	170	140	160	125	248	284	80	85	100	100	65	84.5	278	337	456.5	364.5	7
21	170	140	160	125	248	284	80	85	100	100	65	84.5	278	337	456.5	364.5	871
22	170	140	160	125	248	284	80	85	100	100	65	84.5	278	337	456.5	364.5	851
23	170	140	160	125	256	301	90	95	100	110	70	94.5	286	361	463.5	388.5	8
24	170	140	160	125	256	301	90	95	100	110	70	94.5	286	361	463.5	388.5	9
25	170	140	160	125	256	301	90	95	100	110	70	94.5	286	361	463.5	388.5	9
26																	

Slika 11: ***Design Table u Microsoft Excelu***

Nakon unosa svih dimenzija, iste je potrebno transformirati kako je ranije objašnjeno. Upravo smo taj korak procijenili kao točku u kojoj se gubi najviše vremena i točku koju je potrebno automatizirati. Transformacija kota automatizirana je kombinacijom formula koje spajaju određene ćelije te dviju Macro procedura. Veze između radnih listova prikazanih slikom [9], a samim time i uloge Macro procedura te formula u Excelu, objašnjene su primjerom.

U „Radnom listu sa tablicom za odabir serije vratila“ (3.) nalaze se dvije tipke za



Slika 12: ***Tablica za odabir serije vratila***

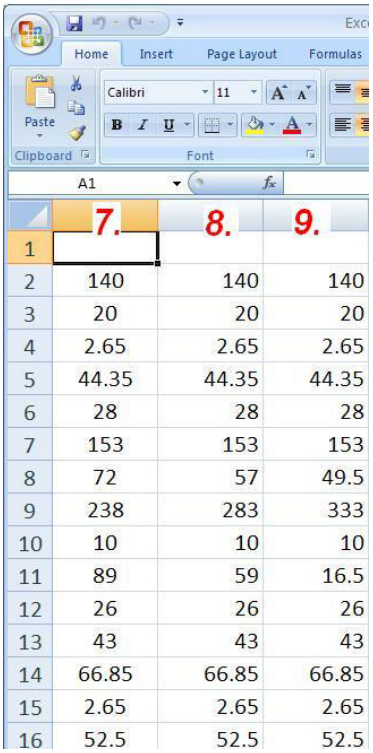
pokretanje Macro procedura te polja za unos granica serije vratila. Postoje dvije opcije kod postupka transformiranja dimenzija i svaka od njih upravljana je svojom Macro procedurom.

1. **Pojedinačni proračun** – Macro procedura transformira dimenzije samo jednog vratila. Pokreće se tipkom „START“ koja s lijeve strane ima polje za unos rednog broja vratila. Jasno je da se ova opcija transformiranja bira kada želimo, metodom prijenosnih matrica, proračunati jedno vratilo.
2. **Proračun serije vratila** – Služi za transformaciju dimenzija odabrane serije vratila. U ovom koraku se, prije transformiranja dimenzija, takoreći definiraju granice serije koja će se proračunati metodom prijenosnih matrica. Raspon vratila za proračun definira se početnom i konačnom granicom koje se unose u zadana polja.

Zadaća gore definiranih Macro procedura je dakle jednaka. Obje imaju ulogu transformiranja dimenzija, samo se razlikuju u količini vratila koja obrađuju. U primjeru pretpostavimo da nam je potrebna analiza sedmog, osmog i devetog vratila u tablici na slici [12]. Radionički crtež sa slijepim kotama za ovakav oblik vratila prikazan je slikom [15]. Oznake na kotama jednake su naslovima stupaca u Design Table-u i tablici za odabir serije vratila što nam olakšava eventualni unos novih vratila u tablicu.

Broj segmenta	Dužina segmenta:		Promjer segmenta:		Radijus zaobljenja:	
1	A	140	ØD	60	r	1.6
2	G-N-A	20	$Ød_1$	64.9		0
3	P	2.65	$Ød_2$	62	0.2	0.2
4	N-(G-C)-P	44.35	$ØD_1$	65	r	1.6
5	E-C	28	$Ød_3$	65.3	r	1.6
6	G-E	153	$ØD_2$	68	r	1.6
7	L-H-G	74.5	$Ød_4$	89		0
8	J-K	228	$ØD_3$	80	0.5	0.5
9	K	10	$Ød_5$	76	0.5	0.5
10	H-J-G_1	96.5	$Ød_6$	79.9		0
11	G_1-E_1	26	$Ød_3$	65.3	r	1.6
12	E_1-C_1	43	$ØD_1$	65	0.5	0.5
13	N_1-(G_1-C_1)-P	66.85	$ØD_5$	64.5	0.5	0.5
14	P	2.65	$Ød_2$	62	0.2	0.2
15	G_1-N_1-A_1	52.5	$Ød_7$	64.4		0
16	A_1	140	$ØD_4$	55	r	1.6

Slika 13: ***Tablica sa transformiranim dimenzijama***



Slika 14: ***Tablica sa transformiranim dimenzijama čitave serije vratila***

Nakon što smo dakle napravili model vratila u CATIA-i, unijeli sve podatke za određenu skupinu vratila u Design Table i tablicu za odabir serije vratila te odredili koja vratila u tablici trebamo proračunat, upisujemo granice u polja lijevo do „START“ tipke te pokrećemo Macro proceduru. U našem slučaju stavljamo u polja brojeve sedam i devet. Pokretanjem procesa Macro procedura prvo briše sve podatke iz „Radnog lista sa odgovarajućim oblikom dimenzija za čitavu seriju vratila“ (5.). Naime postoji mogućnost da su u spomenutom radnom listu ostali podaci iz prethodnih procesa tj. transformacija. Nakon toga procedura kopira čitavi sedmi redak u referentni redak označen na slici [12]. Tablica na „Radnom listu sa transformiranim dimenzijama“ (4.) sastoji se od redaka koji opisuju pojedine segmente. Prvi redak sadrži podatke o prvom segmentu i tako dalje. Ćelije u retku koji opisuje segment i one u referentnom retku su spojene formulama koje izračunavaju duljinu i promjer segmenta te prijelazni radijus. Tako je primjerice jednadžba ćelije F14, koja sadrži duljinu sedmog segmenta (označeno na slici [13]), jednaka

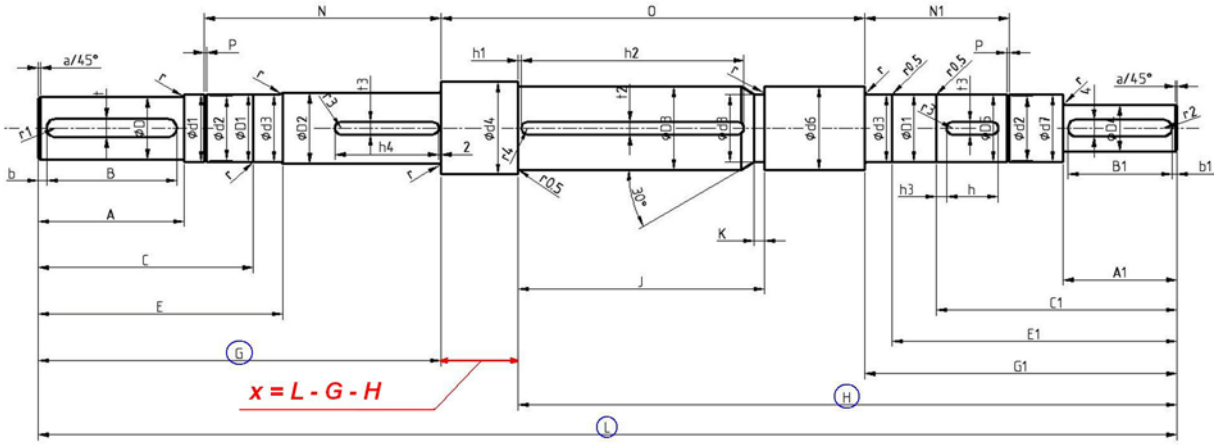
„=Sheet2!AF8 – Sheet2!AA8 – Sheet2!AC8“, {1.}

što je izvedeno iz formule

„L – G – H“ {2.}

za referentni redak u radnom listu (3.) ; „Sheet2“ prikazan slikom [12]. Formula {2.} dobivena je na temelju vratila sa slijepim kotama prikazanog slikom [15]. Kako se trenutačno u referentnom retku nalaze podaci za sedmo vratilo, tablica na „Radnom listu sa transformiranim dimenzijama“ (4.) sadrži dimenzije za sedmo vratilo; slika [14].

Sljedeći korak Macro procedure je kopiranje svih podataka o segmentima za sedmo



Slika 15: ***Shema za transformaciju kota***

vratilo u prvi stupac „Radnog lista sa odgovarajućim oblikom dimenzija za čitavu seriju vratila“ (5.). Kako su dimenzije sedmog vratila transformirane, Macro kopira osmo vratilo u referentni redak i postupak se ponavlja. Jedina razlika je u tome što ovaj puta Macro kopira podatke u drugi stupac u „Radnom listu sa odgovarajućim oblikom dimenzija za čitavu seriju vratila“ (5.). Nakon što je obrađeno i posljednje, deveto vratilo petlja se prekida i Macro završava svoj rad.

MACRO PROCEDURE	
Korak:	Objašnjenje
1.	- priprema radnog lista „Radni list sa odgovarajućim oblikom dimenzija za čitavu seriju vratila“ (5.), prikazanog slikom [14] - procedura očisti sve podatke koji se nalaze unutar radnog lista, kako bi se na isto mjesto mogli smjestiti rezultati nove serije - radni list (5.) služi za spremanje dimenzija segmenata svih vratila iz određene serije i to u obliku pogodnom za analizu - upravo se sa njim kasnije povezuje program napravljen u Vb
2.	- koraci koji su označeni plavom bojom nalaze se unutar gore objašnjene petlje - u drugom koraku procedura kopira podatke iz odabranog retka u referentni redak označen plavom bojom na slici [12] - ćelije u referentnom retku formulama su spojene sa tablicom u radnom listu „Radni list sa transformiranim dimenzijama“ (4.).
3.	- podaci dobiveni u gore navedenoj tablici kopiraju se u radni list (5.) - napomenimo kako svaki stupac u radnom listu (5.) označava jedno vratilo iz serije - nakon ovog koraka slijedi kraj petlje tj. petlja; ili ponavlja osvijetljene radnje za sljedeće vratilo u seriji, ili se prekida
4.	- u posljednjem koraku se resetiraju sve selekcije u radnim listovima - ovaj korak služi za vizualnu dekoraciju .xls datoteke

Tablica 2: ***Macro procedure***

U nastavku je prikazan Macro sa odgovarajućim komentarima, kojima je objašnjen postupak koji obavlja procedura. Za detaljnije objašnjenje je odabran Macro koji obrađuje

seriju vratila, dok je Macro koji služi za obradu samo jednog vratila gotovo identičan pa neće biti posebno analiziran, već će se samo obratiti pažnja na eventualne razlike.

```
      Sub start2_click()
      Dim i, j, n, m As Integer     'definiranje potrebnih varijabli
      i = Cells(4, 5).Value   'zadavanje graničnih vrijednosti
      j = Cells(5, 5).Value
      Sheets("Sheet3").Select       'selektiranje radnog lista „Sheet3“
      Cells.Select      'odabir svih polja u aktivnom radnom listu
      Selection.ClearContents       'brisanje eventualnih podataka
      Sheets("Sheet1").Select
      'Naredna For petlja je jedina razlika između dvaju Macro-a. Ona
ponavlja naredbe koje se nalaze unutar nje za svako od odabranih vratila. U
prvom Macro-u takva petlja nije potrebna zato što se sve naredbe moraju
provesti samo jednom.
      For n = i To j    'brojač n se kreće od donje do gornje zadane
granice
      m = m + 1
      Cells(n + 8, 1).EntireRow.Copy 'odbire redak sa indeksom n + 8 te se
kopira
      Rows("8:8").Select
          ActiveSheet.Paste   'podaci iz odabranog retka se prilijepe u
osmi redak
      'Slijedeći dio služi za prikladan vizualni prikaz tablice (promjena
boje, fonta itd.)
          Application.CutCopyMode = False
          Range("D8:BX8").Select
          With Selection.Interior
              .Pattern = xlSolid
              .PatternColorIndex = xlAutomatic
              .ThemeColor = xlThemeColorAccent5
              .TintAndShade = 0.599993896298105
              .PatternTintAndShade = 0
          End With
          Selection.Font.Bold = True
          Sheets("Sheet2").Select
          Range("F8:F23").Select 'odabir dijela podataka iz tablice u
radnom listu „Sheet 2“
          Selection.Copy
          Sheets("Sheet3").Select
          Cells(2, m).Select  'kopira iste te ih prilijepi na željeno mjesto
                              na radni list „Sheet 3“
          Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone,
SkipBlanks _
              :=False, Transpose:=False
          Sheets("Sheet2").Select
          Range("H8:H23").Select
          Application.CutCopyMode = False
          Selection.Copy
          Sheets("Sheet3").Select
          Cells(18, m).Select
          Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone,
SkipBlanks _
              :=False, Transpose:=False
          Sheets("Sheet2").Select
          Range("J8:J23").Select
          Application.CutCopyMode = False
          Selection.Copy
          Sheets("Sheet3").Select
          Cells(34, m).Select
```

```
            Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone,
SkipBlanks _
                :=False, Transpose:=False
            Sheets("Sheet1").Select
        Next n
        'Ostatak koda služi za resetiranje svih selekcija te vraćanje na
radni list „Sheet 1“
            Sheets("Sheet3").Select
            Range("A1").Select
            Sheets("Sheet2").Select
            Application.CutCopyMode = False
            Range("A1").Select
            Sheets("Sheet1").Select
            Range("E5").Select
        End Sub
```

Kod izrade .xls datoteke za novi tip vratila najbolje je kopirati već postojeću datoteku te izmijeniti tablice unutar nje. Time izbjegavamo izradu novih Macro procedura. Potrebna je dakako i manja izmjena unutar samih Macro procedura, no ona nije suviše zahtjevna. Izmjene se odnose samo na odabir raspona ćelija koje sadrže određene podatke. Dijelovi koje treba izmijeniti podebljani su unutar gore navedenog koda. Jasno je da se na podebljana mjesta unose novi rasponi ćelija, koji su se promijenili zbog novog broja segmenata na vratilu. Nova xls. datoteka zahtjeva i nove relacije između ćelija u referentnom retku i ćelija u tablici prikazanoj slikom [13].

Nakon što smo izvršavanjem jedne od „Start“ naredbi dobili tražene dimenzije segmenata određene serije vratila, preostaje nam samo spremiti i zatvoriti .xls datoteku. Ostale korake u procesu unosa podataka u program obavljamo kroz sučelje istog.

Zaključimo poglavlje o spomenuta dva načina unosa napomenom da se u ovom dijelu unose samo dimenzije vratila. Time se misli na promjer i duljinu svakog segmenta te na prijelazne radijuse. Svi ostali podaci, kao što su podaci o ležajima, opterećenjima i materijalu unose se kroz sučelje napravljeno u Visual Basic-u.

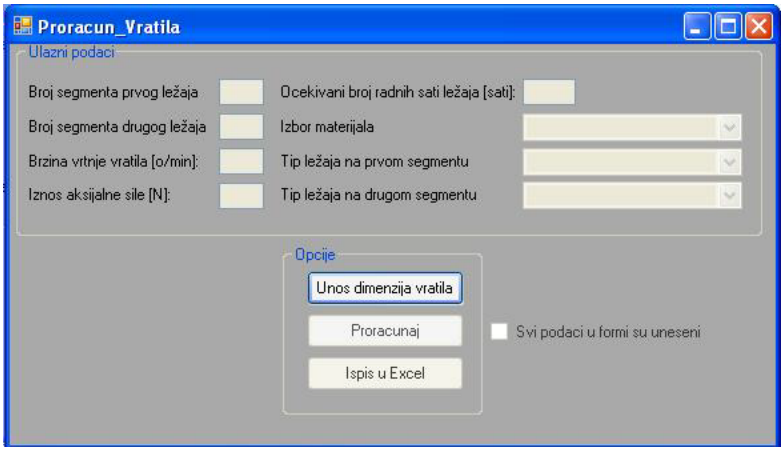
5. UNOS OSTALIH PODATAKA TE PRIPREMA UNESENIH PODATAKA ZA PRORAČUN PRIJENOSNIM MATRICAMA

Prethodnim poglavljem izrada programa raščlanjena je na nekoliko dijelova koje možemo shvatiti kao zasebna poglavlja. Iz tablice [1] vidljivo je da se rad sastoji od unosa podataka, prilagodbe podataka i same analize te od obrade i prikaza izlaznih podataka. Unos podataka može se nadalje podijeliti u kategorije prikazane tablicom [3].

UNOS PODATAKA
- unos dimenzija vratila dobivenih na temelju modela vratila iz CATIA-e   - velik dio ovog unosa objašnjen je u prethodnom poglavlju
- unos dimenzija te ostalih karakteristika ležaja potrebnih za uležištenje vratila   - ovaj dio je također automatiziran i korisnik mora samo referirati program na bazu podataka koji definiraju ležajeve
- unos ostalih podataka potrebnih za analizu vratila (materijal vratila, opterećenja, brzina vrtnje i dr.)   - podaci se unose „ručno“ kroz sučelje programa

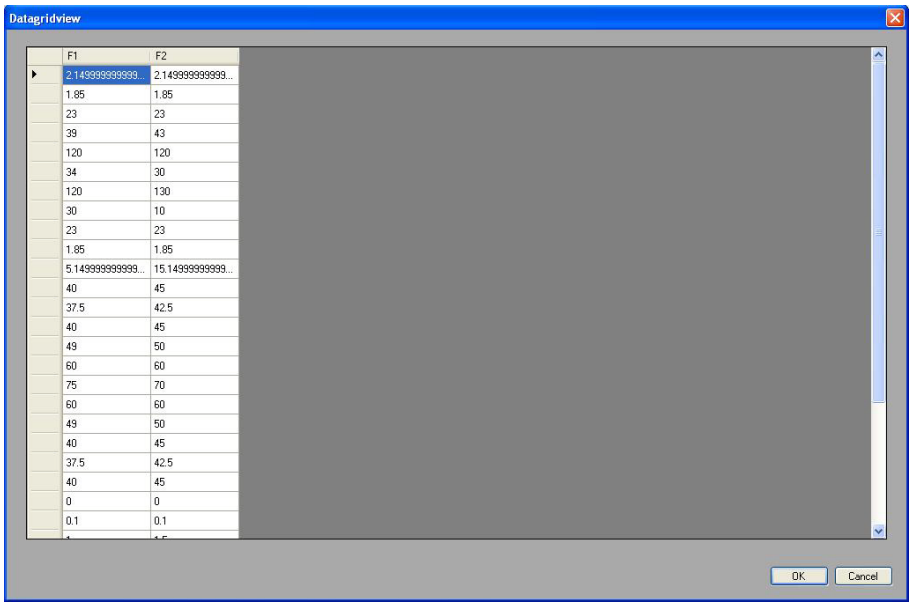
Tablica 3: ***Unos podataka***

Preostalim dijelom unosa tj. unosom koji nije objašnjen u prethodnom poglavlju, upravlja se preko različitih sučelja programa, prikazanih slikama [16], [18] i [19]. Postupak unosa objašnjen je na identičan način kao i u prethodnom poglavlju te se uz primjer objašnjava korak po korak. Važno je napomenuti kako je program izrađen tako da korisnika vodi kroz postupke i ne dozvoljava mu da unese krive informacije. U slučaju unosa informacija kojima nije moguće provesti pravilnu analizu vratila, program se prekida.



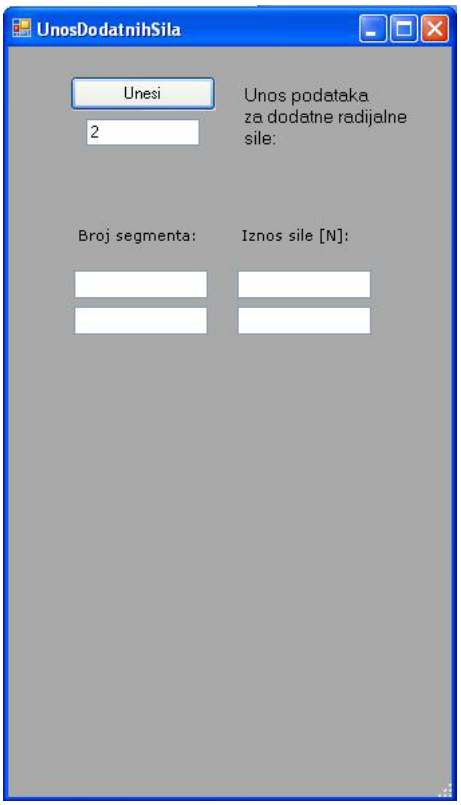
Slika 16: ***Osnovno sučelje glavnog programa***

Pokretanjem glavnog programa otvara se osnovno sučelje prikazano slikom [16]. Prvi korak nakon pokretanja je unos dimenzija vratila iz .xls datoteke pripremljene u prethodnom koraku. Pritiskom na „Unos dimenzija vratila“ otvara se „Open Dialog“ kojim biramo .xls datoteku sa dimenzijama serije vratila. Uspješan unos potvrđuje nam pojavljivanje tablice sa unesenim dimenzijama sl.[17].



Slika 17: ***Vizualni prikaz unesenih dimenzija***

Nakon toga dolazi na red unos ostalih informacija potrebnih za analizu. Kako su se u prethodnom koraku aktivirali prozori za unos informacija o segmentima na kojima se nalaze ležajevi, opcije za izbor materijala i tipa ležaja te ostali prozori na glavnoj formi, slijedi manualno unošenje tih podataka. Primijetimo kako treba, prije pokretanja bilo koje druge opcije, uključiti „Check Box“ kojim se potvrđuje da su uneseni svi potrebni podaci. Uključivanjem istog omogućena je aktivacija opcije „Proračunaj“. Nakon pokretanja opcije „Proračunaj“ javlja se forma prikazana slikom [18], u kojoj prvo unosimo broj segmenata koji su opterećeni radijalnom silom, a kasnije i redne brojeve istih te iznos sila na svakome.

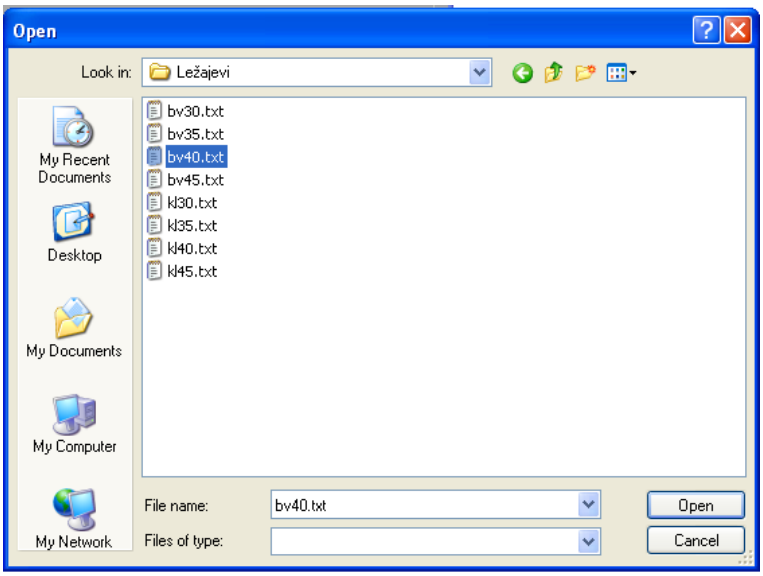


Slika 18: ***Sučelje za unos dodatne radijalne sile***

Slika 19: ***Sučelje za unos torzionog momenta***

Pritiskom na opciju „Unesi“ memoriramo unesene veličine te pokrećemo formu za unos torzionog momenta, vidljivu na slici [19]. U ovom sučelju se unose prvi i posljednji opterećeni segment te iznos torzionog momenta. Obje forme koje se pojavljuju naknadno izrađene su u obliku „Dialog form“, što znači da nisu moguće nikakve druge interakcije sa ostatkom programa dok se one ne ispune i zatvore.

Slijedi referiranje na mapu sa .txt datotekama koje sadrže podatke o ležajevima, o čemu nas obavještava odgovarajuća poruka. Nakon potvrde da je poruka pročitana ponovno se otvara „Open Dialog“ prikazan slikom[20].



Slika 20: ***Open Dialog***

Referiranje na mapu sa ležajevima vrši se na način da se u prikazanom „Open Dialog-u“, odabere bilo koja .txt datoteka sa podacima o ležaju iz mape. Označeni ležaj služi samo kao putanja i nije odabran kao odgovarajući tip ležaja za vratilo. Željeni tip ležaja odabrali smo preko sučelja na slici [16], dok potrebni promjer odabire sam program. Pritiskom na opciju „Open“ unijeli smo i posljednje podatke potrebne za proračun vratila pa program počinje izvršavati ostatak proračuna. Na kraju se javlja poruka koja nas obavještava o uspješno obavljenom proračunu, a istovremeno se aktivira opcija „Ispis u Excel“, kojom spremamo izlazne podatke u .xls datoteku. Pokretanjem opcije „Ispis u Excel“ otvara se sučelje za odabir datoteke u koju želimo spremiti izlazne podatke. U tu svrhu napravljena je .xls datoteka „Izlazni podaci“, koja služi kao osnovni predložak za ispis podataka. Ista je prikazana slikama [23] i [24], a detaljnije je objašnjena u sljedećem poglavlju.

Ako se krene dublje u analizu samog programa, lako je za primijetiti kako se on sastoji od više procedura koje obavljaju zasebne zadaće. U nastavku će biti objašnjene osnovne zadaće procedura unutar programa i to redoslijedom njihove aktivacije. Ispis svih procedura predočen je tablicom [4].

POPIS PROCEDURA UNUTAR PROGRAMA		
1.	`BtnUnosdimenzija_Click`	- proces služi za unos dimenzija vratila iz Excel datoteke u program
2.	`BtnProracunaj_Click`	- pokreće osnovni proces koji upravlja većinom ostalih
3.	`ObradaIzboraMaterijala()`	- unos podataka o materijalu
4.	`UnosDodatnihSila.ShowDialog()`	- unos dodatnih radijalnih sila kroz posebno sučelje
5.	`UnosTorMomenta.ShowDialog()`	- unos torzionog momenta te područja djelovanja istog
6.	`odabirbrojaradnihsati()`	- unos očekivanog broja radnih sati za ležaje
7.	`PomocZaOdabirLezaja()`	- definiranje putanje za mapu sa ležajima
8.	`odabirlezaja()`	- odabir potrebnih ležaja te učitavanje podataka koji ih opisuju
9.	`PodjelaVratilaNaPolja()`	- podjela segmenata na polja te izračun mase momenta torzije i dr.
10.	`ZarezniFaktor1() i ZarezniFaktor2()`	- definicija zareznih faktora kod savijanja i uvijanja vratila
11.	`PreuzetiDio()`	- kod za proračun metodom prijenosnih matrica
12.	`ProracunLezajaL() i ProracunLezajaD()`	- proračun vijeka trajanja ležaja te dinamičkog opterećenja ležaja
13.	`DefinicijaC0iC()`	- definicija statičke i dinamičke nosivosti ležaja
14.	`BtnIspisUExcel_Click`	- proces vrši ispis podataka u Excel datoteku
15.	`VelicineZaDijagram()`	- određuje koordinate polja na osi apscisa u dijagramima izlaznih podataka
16.	`ProracunVratila_FormClosed`	- gasi sve zaostale EXCEL.EXE datoteke u Processes direktoriju
17.	`CheckBox1_CheckedChanged`	- osigurava pravilan redoslijed rada na glavnoj formi

Tablica 4: ***Popis procedura unutar programa***

BtnUnosdimenzija_Click pokreće se pritiskom na opciju „Unos dimenzija vratila“. Procedura otvara vezu sa Microsoft Excel-om te povlači podatke o vratilu iz odabrane datoteke. Prikazuje uspješan unos u DataGridView-u te ih sprema u, za tu svrhu namijenjena, polja. Naposljetku regulira aktivnost polja za unos teksta i drugih opcija na glavnoj formi.

Napomenimo još jednom da se pod aktivnošću misli na mogućnost biranja određene opcije ili unosa teksta u određeno polje za unos teksta.

d_s(500, 50) – polje sadrži duljine segmenata svih vratila u unesenoj seriji
pro(500, 50) – polje sadrži promjere segmenata svih vratila u unesenoj seriji
rad(500, 50) – polje sadrži radijus na prijelazu segmenata svih vratila u unesenoj seriji

Prva dimenzija u polju, dimenzija koja ima 500 slobodnih mjesta, označava indeks segmenta na vratilu, a druga dimenzija označava indeks vratila. Indeks u polju se može objasniti kao redni broj, ali mu je početna vrijednost 0, a ne 1.

BtnProracunaj_Click je glavna procedura čitavog programa. Pokreće se biranjem opcije „Proračunaj“, a upravlja svim procedurama osim unosa i ispisa. Ona poziva većinu procedura navedenih u tablici [4]. Nakon poziva procedure za unos ostalih podataka tj. podataka koji nisu uneseni procedurom „BtnUnosdimenzija_Click“, pokreće dvije gotovo najvažnije petlje u cjelokupnom programu. Prva omogućava proračun serije vratila, a druga automatski odabir šireg ležaja, ukoliko dinamička nosivost prethodnog ne zadovoljava zahtjeve. „BtnProracunaj_Click“ također regulira aktivnost polja za unos i opcija za pokretanje procedura.

ObradaIzboraMaterijala() je prva procedura koju pokreće „BtnProracunaj_Click“. Na temelju odabranog čelika ona definira glavna svojstva istog. Čelik se bira „ComboBox-om“ „Očekivani broj radnih sati ležaja“. Podaci o materijalu su izvađeni iz tehničke literature [6].

Gdn – dinamička izdržljivost na vlak/tlak
Gfdn – dinamička izdržljivost na savijanje pri kolebanju opterećenja sa pozitivne na negativnu maksimalnu vrijednost
Tdi – dinamička izdržljivost na torziju pri kolebanju opterećenja između maksimalne vrijednosti i nule

UnosDodatnihSila.ShowDialog() pokreće zasebnu formu kroz koju se unose dodatne radijalne sile na vratilo. Izgled forme prikazan je slikom [18], a sama forma je popraćena kodom koji regulira unos. Prvo se, na temelju unesenog broja u polje ispod opcije „Broj segmenata“ (kasnije „Unesi“), definira broj segmenata sa dodatnim radijalnim opterećenjem. Pritiskom na tipku „Broj segmenata“ stvara se ranije odabrani broj redaka za unos podataka o dodatnim silama. Istovremeno se naslov tipke „Broj segmenata“ mijenja u „Unesi“. Svaki od novonastalih redaka ima dva polja za unos teksta, od kojih je jedan namijenjen za unos rednog broja, a drugi za iznos sile. Uneseni podaci spremaju se u polja „a(100)“ i „b(100)“ pritiskom na tipku „Unesi“. Unos je također ograničen na način da ne dozvoljava korisniku unošenje teksta, već samo brojeve.

a(100) – polje u koje se sprema redni broj segmenata opterećenih dodatnom radijalnom silom
b(100) – polje sa iznosima radijalne sile na segmentima definiranima u polju a(100)

UnosTorMomenta.ShowDialog() je naredba vrlo slična prethodnoj, a time se misli na činjenicu da ona također pokreće zasebnu formu. Forma za unos torzionog momenta prikazana je slikom [19], a njezin kod služi za definiranje triju varijabli koje opisuju torzioni moment. Ovim korakom dakle definiramo prvi i posljednji segment koji su opterećeni torzionim momentom te iznos samog.

Odabirbrojaradnihsati(). Procedura koja učitava broj radnih sati, a istovremeno provjerava da li je unesena veličina u pravom obliku i brojčanom, a ne tekstualnom. Broj radnih sati se unosi u polje „Očekivani broj radnih sati ležaja“.

PomocZaOdabirLezaja() je posljednja procedura koja se izvodi izvan petlje za proračun više vratila. Njezina je zadaća da definira putanju mape u kojoj se nalaze datoteke sa podacima o ležajevima. Nakon što korisnik, kroz „Open Dialog“, odabere jedan od ležaja u mapi, program mijenja putanju ležaja u putanju čitave mape. Kasnije se, u proceduri odabirlezaja(), toj putanji pridodaje ležaj definiran odabranim tipom i promjerom segmenta.

Odabirlezaja(). Kao što je ranije objašnjeno, procedura Odabirlezaja() prvo definira putanju ležaja potrebnog za određeno vratilo u seriji. Putanja se dobiva zbrajanjem putanje mape sa datotekama i naziva potrebne datoteke. Svaka .txt datoteka imenovana je na specifičan način. Njeno ime sastoji se od skraćenice za tip ležaja te promjera ležaja. Odgovarajuće skraćenice za određene tipove ležaja navedene su u tablici [5].

VRSTE LEŽAJA	
kl -	Kruti radijalni ležaj (kuglični)
cv -	Cilindrično valjkasti ležaj
il -	Igličasti ležaj
bv -	Bačvasto valjkasti ležaj
akl -	Aksijalni kuglični ležaj
avl -	Aksijalni valjkasti ležaj

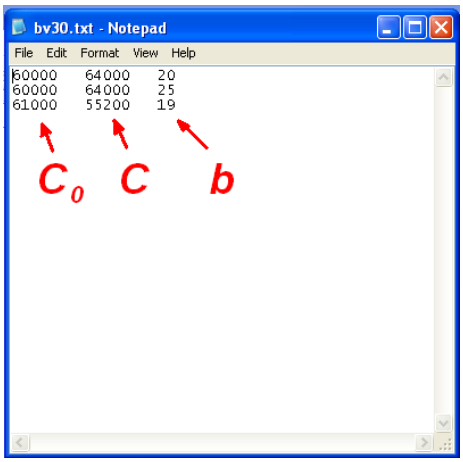
Tablica 5: ***Vrste ležaja***

PUTANJA MAPE + NAZIV .TXT DATOTEKE = PUTANJA LEŽAJA
F:\Diplomski\Ležajevi\ + bv30.txt = F:\Diplomski\Ležajevi\bv30.txt

Skraćenice iz tablice [5] definiraju se preko osnovnog sučelja [16], a promjer se određuje na temelju promjera segmenta na kojem je ležaj. Procedura nadalje otvara definirane .txt datoteke te iz njih izvlači informacije o ležajevima, koje sprema u zadana polja.

sila_l(100) – dinamička nosivost (C_0)
stsila_l(100) – statička nosivost (C)
f_l(100) – širina ležaja (b)

Kod izrade novih .txt.datoteka sa podacima za ležaj treba unositi podatke na način prikazan slikom [21]. Također je važno napomenuti da u datoteci ne smije biti ni jednog praznog retka.



Slika 21: ***.txt datoteka za ležaj***

PodjelaVratilaNaPolja(). Za proračun vratila prijenosnim matricama potrebno je vratilo podijeliti na što je moguće više polja tj. diskova. Naime svako polje/disk na vratilu označava zasebni linearni račun u prijenosnim matricama. Upravo je za to namijenjena procedura PodjelaVratilaNaPolja(). Njena osnovna zadaća je da na temelju duljine segmenta isti podijeli na odgovarajući broj polja. Odnos duljina i broj polja predočen je tablicom [6].

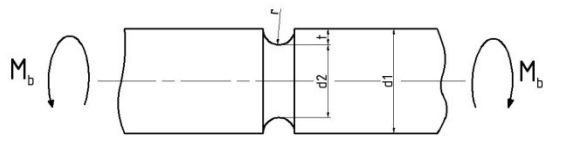
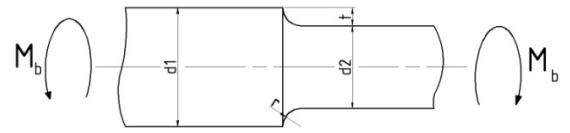
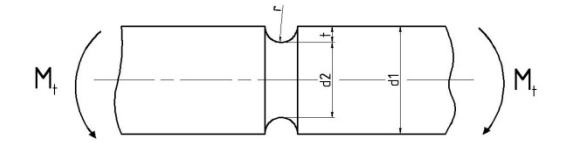
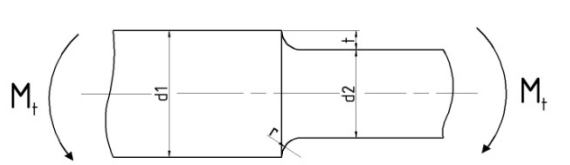
d_s – duljina segmenta	n – broj polja
d_s <= 20	3
20 < d_s <=40	5
40 < d_s <= 100	8
100 < d_s	12

Tablica 6: ***Raspodjela segmenata na polja***

Napomenimo kako su segmenti na kojima su ležajevi razrađeni na drugačiji način. Oni se sastoje od jednog polja širine ležaja, a preostali dio je podijeljen prema formuli iz tablice [6]. Procedura, osim što dijeli vratilo na polja, istovremeno izračunava masu i moment inercije svakog polja. Definira koja su od novonastalih polja, opterećena dodatnim silama i torzionim momentom. Sve dobivene informacije pohranjuju se u predviđena polja. Prvi indeks označava polje, a drugi vratilo u seriji.

p_s(500, 50) – duljina polja
pr_seg(500, 50) – promjer polja
mi(500, 50) – moment inercije određenog polja
masa(500, 50) – masa polja
bool(500, 50) – Boolean varijabla. Vrijednost „True" dobiva na poljima koja su opterećena torzionim momentom.
fdod(500, 50) – iznos dodatne sile na određenom polju
p11(50) – broj polja u čitavom vratilu
l1- indeks polja na kojemu se nalazi prvi ležaj
l2 - indeks polja na kojemu se nalazi drugi ležaj

ZarezniFaktor1() i ZarezniFaktor2() su procedure koje određuju faktore zareznog djelovanja kod savijanja i uvijanja vratila. Faktori su ovisni o odnosu promjera susjednih segmenata te radijusu prijelaznog zaobljenja. Način na koji se računaju faktori prikazan je tablicom [7], u kojoj se nalaze slike prijelaza i njima pripadajuće formule.

ZAREZNI FAKTORI	
	$\alpha_k = 1 + \frac{1}{\sqrt{\frac{0.25}{t/r} + 1.80\frac{(1+a/r)^2}{(a/r)^3}}}$
	$\alpha_k = 1 + \frac{1}{\sqrt{\frac{0.77}{t/r} + 3.80\frac{(1+a/r)^2}{(a/r)^3} + \frac{0.2}{(t/r)^3}\cdot\frac{a/r}{(a/r+t/r)}}}$
	$\alpha_k = 1 + \frac{1}{\sqrt{\frac{1.0}{t/r} + 7.0\frac{(1+a/r)^2}{(a/r)^3}}}$
	$\alpha_k = 1 + \frac{1}{\sqrt{\frac{3.4}{t/r} + 13.0\frac{(1+a/r)^2}{(a/r)^3} + \frac{1.0}{(t/r)^2}\cdot\frac{a/r}{(a/r+t/r)}}}$

Tablica 7: ***Zarezni faktori***

Iznosi faktora zareznog djelovanja pohranjeni su u polja lk i lk2.

lk(500,50) – faktor zareznog djelovanja kod savijanja
lk2(500,50) – faktor zareznog djelovanja kod uvijanja

PreuzetiDio() je procedura koja sadrži kod za analizu polja prijenosnim matricama. Ovaj dio možemo shvatiti kao jezgru programa i svi ulazni podaci su pripremljeni za ovu proceduru. Prije same analize potrebno je povezati prethodne varijable sa varijablama iz procedure PreuzetiDio(). Naime tokom izrade programa odabir naziva za određene varijable nije bio jednak kao u postojećem kodu koji se nalazi u proceduri PreuzetiDio(). Nakon izjednačavanja varijabli program računa poprečnu silu, moment savijanja, progib i nagib svakog polja na vratilu metodom prijenosnih matrica. Kako se sistem diferencijalnih jednadžbi prevodi u sistem linearnih jednadžbi, ova metoda daje samo približne rezultate, a točnost rezultata ovisi o broju polja na koja je podijeljeno vratilo. No za potrebe usporedbe serije vratila koja se analiziraju nama su takvi rezultati zadovoljavajući.

ProracunLezajaL() i ProracunLezajaD() su procedure kojima izračunavamo dinamičku opterećenost odabranih ležaja te vijek trajanja ležaja. Procedura „ProracunLezajaL()“ služi za proračun ležaja na prvom segmentu, a procedura „ProracunLezajaD()“ na drugom. Potrebne faktore za izračun dinamičkog opterećenja dobivamo iz ručno unesene tablice.

DefinicijaC0iC() je kratka procedura koja definira te sprema iznose statičke i dinamičke nosivosti ležajeva za čitavu seriju vratila.

IspisCl(50) – Dinamička nosivost ležaja na prvom segmentu. Indeks polja označava indeks vratila u seriji koja se analizira.
IspisCd(50) – Dinamička nosivost ležaja na drugom segmentu.
IspisC0l(50) – Statička nosivost ležaja na prvom segmentu.

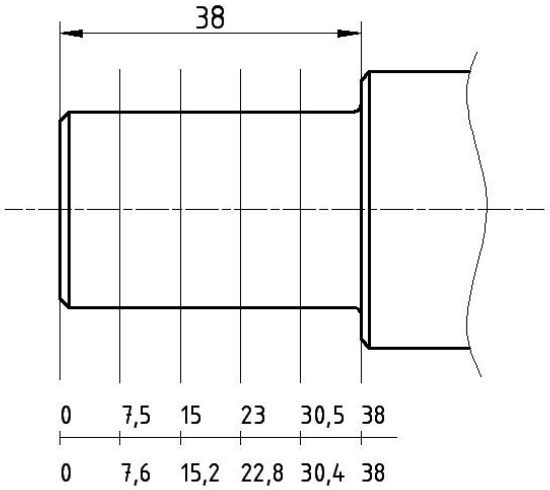
IspisC0d(50) – Statička nosivost ležaja na drugom segmentu.

BtnIspisUExcel_Click pokrećemo pritiskom na opciju „Ispis u Excel“ u glavnom sučelju [16]. Procedura prvo stvara veze sa Microsoft Excelom i željenom .xls datotekom. U ovom slučaju treba odabrati datoteku „IzlazniPodaci“ jer nam ona služi kao predložak za ispis. Nakon toga poziva proceduru „VelicineZaDijagram()“, stvara nove radne listove u .xls datoteci te sprema sve podatke u iste. Datoteka „IzlazniPodaci“ se automatski otvara te korisnik može pregledati ispis. Zatvaranjem osnovne forme zatvara se i datoteka „IzlazniPodaci“.

VelicineZaDijagram(). Problem, zbog kojeg je potrebna procedura „VelicineZaDijagram()“ i svrha iste, biti će u potpunosti objašnjeni sljedećem poglavlju. U ovom trenutku je dovoljno reći da ista stvara polja „XOs(2000,50)“ i „ca(1000, 50)“.

XOs(2000,50) – Polje koje sadrži točke na osi apscisa za dijagrame u datoteci „IzlazniPodaci“. Točke kreću od veličine 0 te rastu za 0,5 do duljine vratila.
ca(1000, 50) – Polje sadrži točke na vratilu koje označavaju polja na koja je vratilo podijeljeno. Točke u polju „ca“ su zaokružene na najbližu veličinu iz polja „XOs“.

Kako bi bilo jasnije, što polja „XOs“ i „ca“ sadržavaju naveden je primjer. Slikom [22] prikazan je prvi segment vratila namijenjenog za analizu.



Slika 22: ***Aproksimacija polja***

Pretpostavimo da se na ovom segmentu ne nalazi ležaj. Kako je duljina segmenta 38mm program će ga podijeliti na pet jednakih dijelova. Koordinate polja u točnom prikazu, prikazane su donjim brojevnim pravcem na slici [22], dok su aproksimirane koordinate prikazane gornjim. Vidljivo je kako su aproksimirane koordinate zaokružene na vrijednost koja je višekratnik broja 0,5. Upravo se takve aproksimirane koordinate spremaju u polje „ca“ te se kasnije koriste kao indeks kojim određujem položaj iznosa naprezanja na osi apscisa.

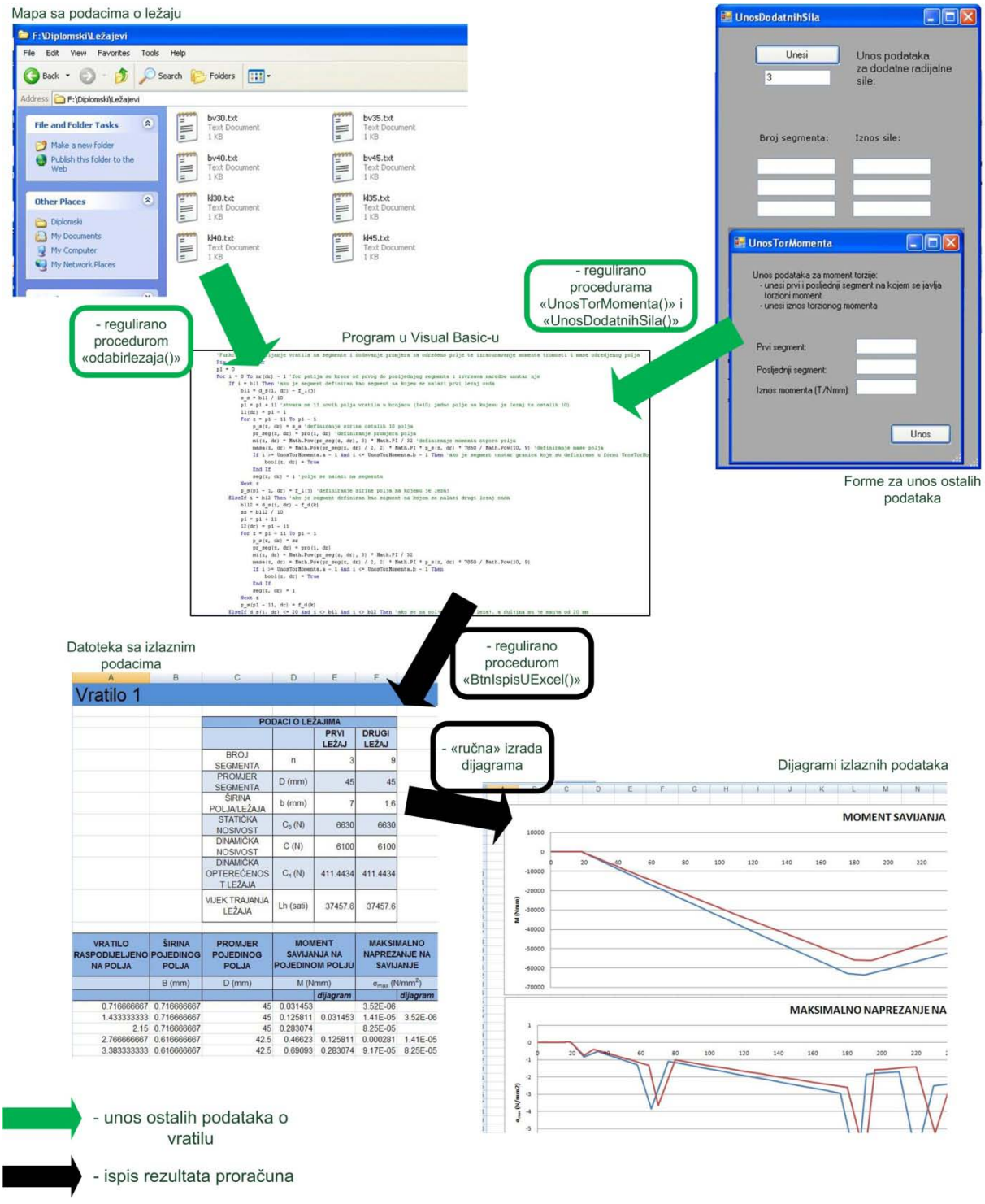
ProracunVratila_FormClosed je procedura koja zatvara sve zaostale „excel.exe“ izvršne datoteke, aktivirane programom. Naime pokretanjem programa i njegovim kasnijim gašenjem aktivirali su se procesi „excel.exe“ koji se kasnije nisu isključili. Traženjem adekvatnog rješenja za ovaj učestali problem na internetu [7] je pronađena procedura koja rješava isti.

CheckBox1_CheckedChanged aktivira i deaktivira tipku „Proračunaj“, nalazi se na osnovnom sučelju [16], ovisno o tome da li je CheckBox uključen ili ne. Time sprječavamo

odabir tipke prije nego su uneseni svi potrebni podaci o vratilu. Korisnik dakle mora prethodno definirati:

- Broj segmenta prvog ležaja
- Broj segmenta drugog ležaja
- Brzina vrtnje vratila
- Iznos aksijalne sile
- Očekivani broj radnih sati ležaja
- Izbor materijala
- Tip ležaja na prvom segmentu
- Tip ležaja na drugom segmentu.

Nakon toga odabire se „Svi podaci u formi su uneseni“, što nam omogućava pritisak na tipku „Proračunaj“. Proces unosa ostalih dimenzija te ispisa rezultata u koracima je predočen slikom [23].

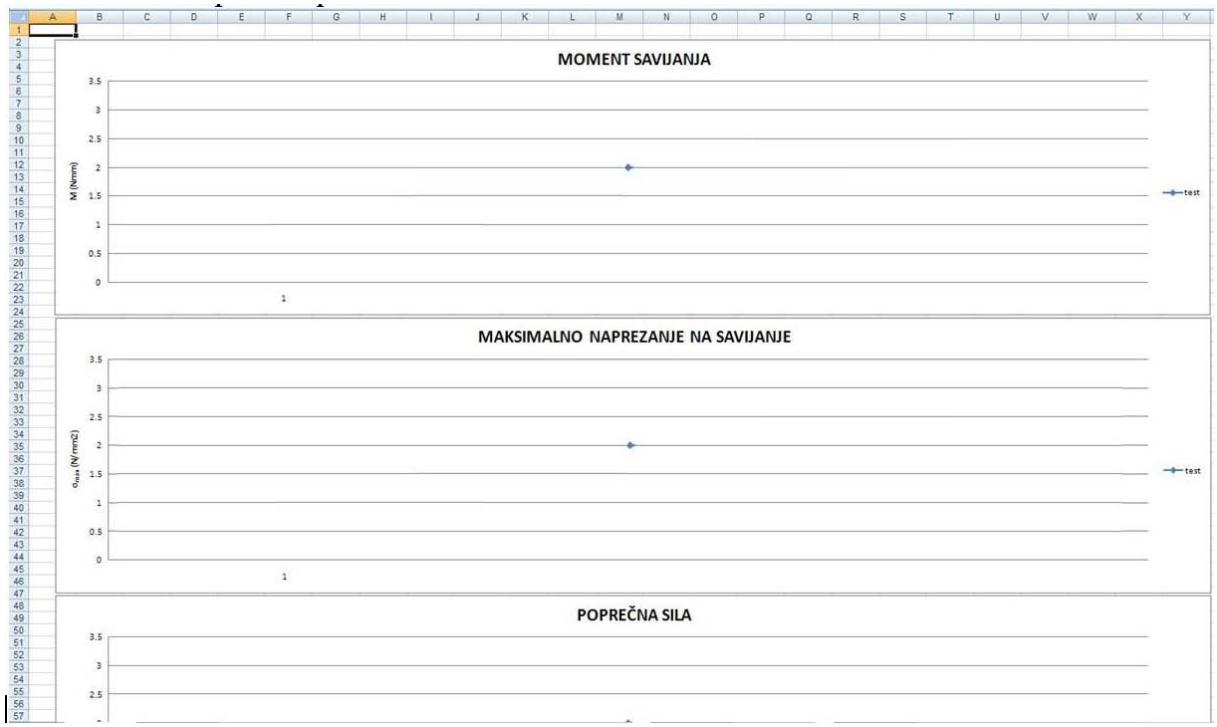


Slika 23: ***Unos ostalih podataka te ispis rezultata***

6. PRIPREMA I OBRADA IZLAZNIH PODATAKA

Već je ranije spomenuto kako je datoteka za ispis izlaznih podataka unaprijed uređena kao predložak. Predložak „IzlazniPodaci“ sastoji se od dva radna lista. Prvi je nazvan „Dijagrami“ i sadrži zajedničke dijagrame opterećenja i naprezanja za željenu seriju vratila. Radni list sa dijagramima u koje još nisu unesena vratila, već sadrži samo „test“ krivulju, prikazan je slikom [24]. Test krivulja služi kako bi na dijagramima bile vidljive veličine na osima te ostale oznake poput naslova legenda i sl. Na temelju toga utvrđujemo koji se podaci unose u koji dijagram (moment,sila i sl.).

Drugi radni list ima istu formu kao radni list „Vratilo1“, prikazan slikom [25], ali nema nikakvih upisanih podataka.



Slika 24: ***Radni list sa dijagramima***

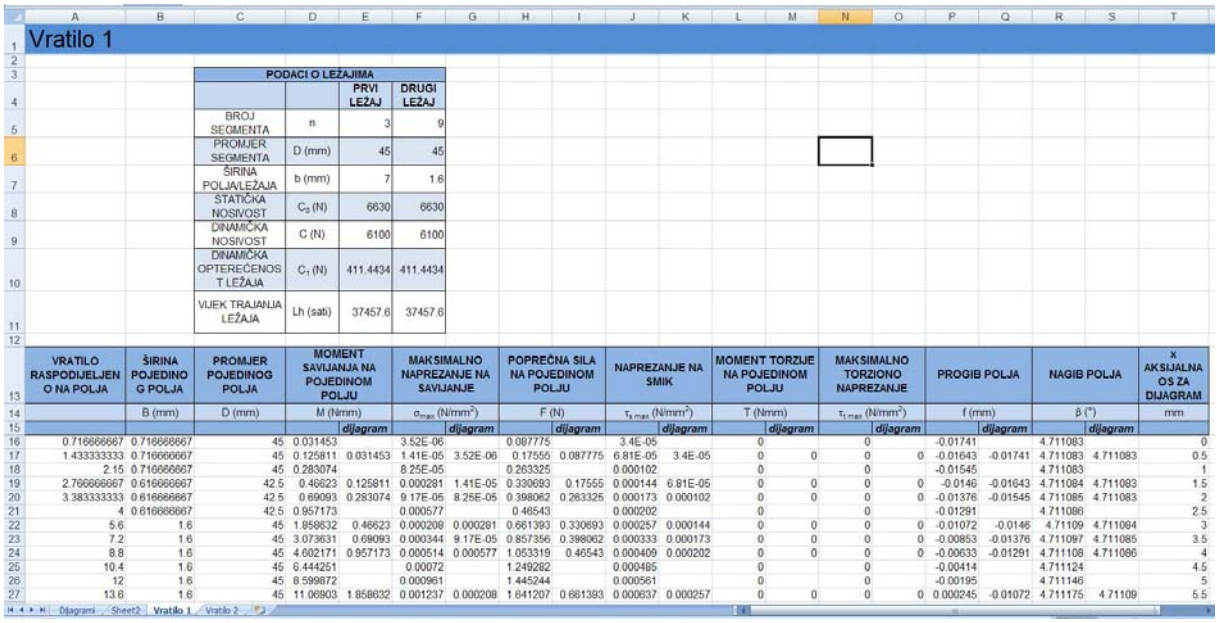
Što se tiče samih izlaznih podataka, oni se mogu podijeliti u dvije skupine. Veći dio podataka opisuje vratilo, dok se ostatak odnosi na opis ležajeva na vratilu. Tablicom [8] prikazani su svi izlazni podaci koje dobivamo ispisom.

IZLAZNI PODACI VRATILA	
Vratilo raspodijeljeno na polja [mm]	- stupac „Vratilo raspodijeljeno na polja“ sadrži koordinate polja vratila na osi apscisa - ovo su stvarni iznosi, a ne aproksimirani (donji brojevni pravac na slici [22])
Širina pojedinog polja [mm]	- sadrži širine svih polja na koja je vratilo podijeljeno
Promjer pojedinog polja [mm]	- promjeri polja na koje je vratilo podijeljeno
Moment savijanja na pojedinom polju [Nmm]	- iznos momenta na savijanje na određenom polju
Maksimalno naprezanje na savijanje	- iznos maksimalnog naprezanja na

[N/mm^2]	određenom polju
Poprečna sila na pojedinom polju [N]	- iznos poprečne sile na određenom polju
Naprezanje na smik [N/mm^2]	- iznos naprezanja na smik na određenom polju
Moment torzije na pojedinom polju [Nmm]	- iznos momenta torzije za određeno polje
Maksimalno torziono naprezanje [N/mm^2]	- iznos maksimalnog torzionog naprezanja na određenom polju
Progib polja [mm]	- progib polja
Nagib polja [°]	- nagib polja
x aksijalna os za dijagram [mm]	- stupac sadrži sve koordinate osi apscisa u dijagramima na radnom listu „Dijagrami“ - N = {0,0.05,1,1.5,…,“duljina vratila“}
PODACI LEŽAJA	
Broj segmenta	- ispis rednih brojeva segmenata na kojima se nalaze prvi i drugi ležaj
Promjer segmenta [mm]	- promjer segmenata na kojima se nalaze ležajevi
Širina polja/ležaja [mm]	- širina polja na kojima se nalaze segmenti
Statička nosivost [N]	- statička nosivost ležajeva
Dinamička nosivost [N]	- dinamička nosivost ležajeva
Dinamička opterećenost ležaja [N]	- dinamičko opterećenje ležajeva
Vijek trajanja ležaja [sati]	- izračunati vijek trajanja ležajeva

Tablica 8: ***Izlazni podaci***

Određene skupine podataka iz tablice [8] ispisane su na dva različita načina što je vidljivo na slici [25]. Prvi stupac je „normalna“ inačica gdje svaka ćelija označava jedno polje vratila. Stupac sa naslovom „dijagram“ sadrži iste podatke kao i prvi samo što su ispisani na drugačiji način. Oni su naime u istom retku kao i aproksimirana koordinata odgovarajućeg polja na osi apscisa. Upravo objašnjenim ispisom u dva stupca, ispisuju se podaci koji se kasnije unose u dijagrame, a to su redom: moment savijanja, maksimalno naprezanje na savijanje, poprečna sila, naprezanje na smik, moment torzije, maksimalno torziono naprezanje, progib te naposljetku nagib.



Vratilo 1

PODACI O LEŽAJIMA		PRVI LEŽAJ	DRUGI LEŽAJ
BROJ SEGMENTA	n	3	9
PROMJER SEGMENTA	D (mm)	45	45
ŠIRINA POLJA/LEŽAJA	b (mm)	7	1.6
STATIČKA NOSIVOST	C_0 (N)	6630	6630
DINAMIČKA NOSIVOST	C (N)	6100	6100
DINAMIČKA OPTEREĆENOST LEŽAJA	C_1 (N)	411.4434	411.4434
VIJEK TRAJANJA LEŽAJA	Lh (sati)	37457.6	37457.6

VRATILO RASPODIJELJENO NA POLJA	ŠIRINA POJEDINOG POLJA	PROMJER POJEDINOG POLJA	MOMENT SAVIJANJA NA POJEDINOM POLJU		MAKSIMALNO NAPREZANJE NA SAVIJANJE		POPREČNA SILA NA POJEDINOM POLJU		NAPREZANJE NA SMIK		MOMENT TORZIJE NA POJEDINOM POLJU		MAKSIMALNO TORZIONO NAPREZANJE		PROGIB POLJA		NAGIB POLJA		x AKSIJALNA OS ZA DIJAGRAM
	B (mm)	D (mm)	M (Nmm)		σ_{max} (N/mm²)		F (N)		$\tau_{s\,max}$ (N/mm²)		T (Nmm)		$\tau_{t\,max}$ (N/mm²)		f (mm)		β (°)		mm
				dijagram		dijagram		dijagram		dijagram		dijagram		dijagram		dijagram		dijagram	
0.716666667	0.716666667	45	0.031453		3.52E-06		0.087775		3.4E-05		0		0		-0.01741		4.711083		0
1.433333333	0.716666667	45	0.125811	0.031453	1.41E-05	3.52E-06	0.17555	0.087775	6.81E-05	3.4E-05	0	0	0	0	-0.01643	-0.01741	4.711083	4.711083	0.5
2.15	0.716666667	45	0.283074		8.25E-05		0.263325		0.000102		0		0		-0.01545		4.711083		1
2.766666667	0.616666667	42.5	0.46623	0.125811	0.000281	1.41E-05	0.330693	0.17555	0.000144	6.81E-05	0	0	0	0	-0.0146	-0.01643	4.711084	4.711083	1.5
3.383333333	0.616666667	42.5	0.69093	0.283074	9.17E-05	8.25E-05	0.398062	0.263325	0.000173	0.000102	0	0	0	0	-0.01376	-0.01545	4.711085	4.711083	2
4	0.616666667	42.5	0.957173		0.000577		0.46543		0.000202		0		0		-0.01291		4.711086		2.5
5.6	1.6	45	1.859832	0.46623	0.000208	0.000281	0.661393	0.330693	0.000257	0.000144	0	0	0	0	-0.01072	-0.0146	4.71109	4.711084	3
7.2	1.6	45	3.073631	0.69093	0.000344	9.17E-05	0.857356	0.398062	0.000333	0.000173	0	0	0	0	-0.00853	-0.01376	4.711097	4.711085	3.5
8.8	1.6	45	4.602171	0.957173	0.000514	0.000577	1.053319	0.46543	0.000409	0.000202	0	0	0	0	-0.00633	-0.01291	4.711108	4.711086	4
10.4	1.6	45	6.444251		0.00072		1.249282		0.000485		0		0		-0.00414		4.711124		4.5
12	1.6	45	8.599872		0.000961		1.445244		0.000561		0		0		-0.00195		4.711146		5
13.6	1.6	45	11.06903	1.858832	0.001237	0.000208	1.641207	0.661393	0.000637	0.000257	0	0	0	0	0.000245	-0.01072	4.711175	4.71109	5.5

Dijagrami Sheet2 Vratilo 1 Vratilo 2

Slika 25: ***Radni list sa izlaznim podacima za jedno vratilo***

U ranijim poglavljima je spomenuto kako proces ispisa podataka u .xls datoteku pokrećemo opcijom „Ispis u Excel“. Isto je tako poznato da time pokrećemo proceduru „BtnIspisUExcel_Click“ koja je objašnjena u prethodnom poglavlju. U narednom dijelu ćemo upotpuniti priču o ispisu podataka te objasniti zašto nam je potrebna pomoćna procedura „VelicineZaDijagram()“.

Nakon analize serije vratila potrebno je rezultate analize za svako vratilo spremiti na odgovarajuće mjesto. Najjednostavnije, a istovremeno najpreglednije rješenje, je napraviti odgovarajući radni list za svako od vratila u seriji te kasnije u njega spremiti podatke. Upravo je to prvi korak kojeg procedura izvodi nakon otvaranja osnovnog .xls predloška i pokretanja procedure „VelicineZaDijagram()“. Kako bi ispis podataka bio pregledan, procedura umjesto da napravi novi prazan radni list, kopira već postojeći, referentni, te ga naziva „VratiloX“. Varijabla „X“ označava redni broj vratila u xls. datoteci. Primjerice; ako su analizirana dva vratila, procedura kopira referentni radni list dva puta te kopije naziva „Vratilo1“ i „Vratilo2“, što je slučaj na slici [25] U novonastale radne listove spremaju se podaci za određeno vratilo. Na temelju dobivenih veličina u stupcima slijedi „ručno“ dodavanje karakteristika vratila u dijagrame koje će biti objašnjeno kasnije. Termin „ručno“ znači da unos podataka u dijagrame nije upravljan programom.

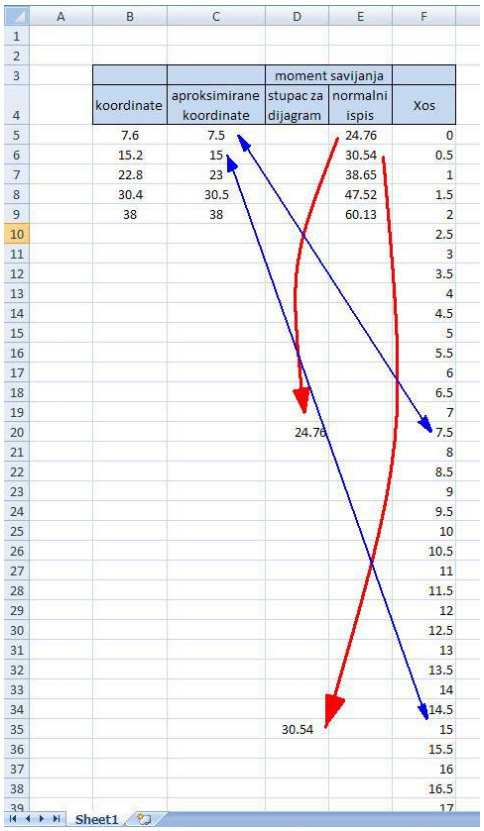
Izrada dijagrama sa prikazom istih karakteristika serije vratila na zajedničkom dijagramu iznjedrila je problem koji zahtijeva pomoćnu proceduru „VelicineZaDijagram()“. Naime raspored i širina polja na vratilu ovisi o duljinama segmenata na određenom vratilu. Kako su u određenoj seriji dimenzije vratila različite, dolazi do razlika u podjeli na polja. Rezultat toga su dva vratila koja imaju različit raspored i širinu polja što nam onemogućava odabir stupca „Vratilo raspodijeljeno na polja“ za os apscisu.

Procedura „VelicineZaDijagram()“ izrađuje novo polje podataka „XOs“ koje služi kao os apscisa na dijagramima.

$$XOs = \{0{,}0.5{,}1{,}1.5{,}2{,}2.5{,}\dots{,}A\},\ \text{A- duljina vratila}$$

Osim polja „XOs“ procedura izrađuje i polje „ca“ koje sadrži aproksimirane koordinate od polja na koja je podijeljeno vratilo. Takve koordinate imaju iste vrijednosti kao pojedine veličine u polju “XOs“ zato što su, kao što je prije objašnjeno, zaokružene na višekratnik broja 0,5. Time je omogućena izrada zajedničkih dijagrama za analiziranu seriju

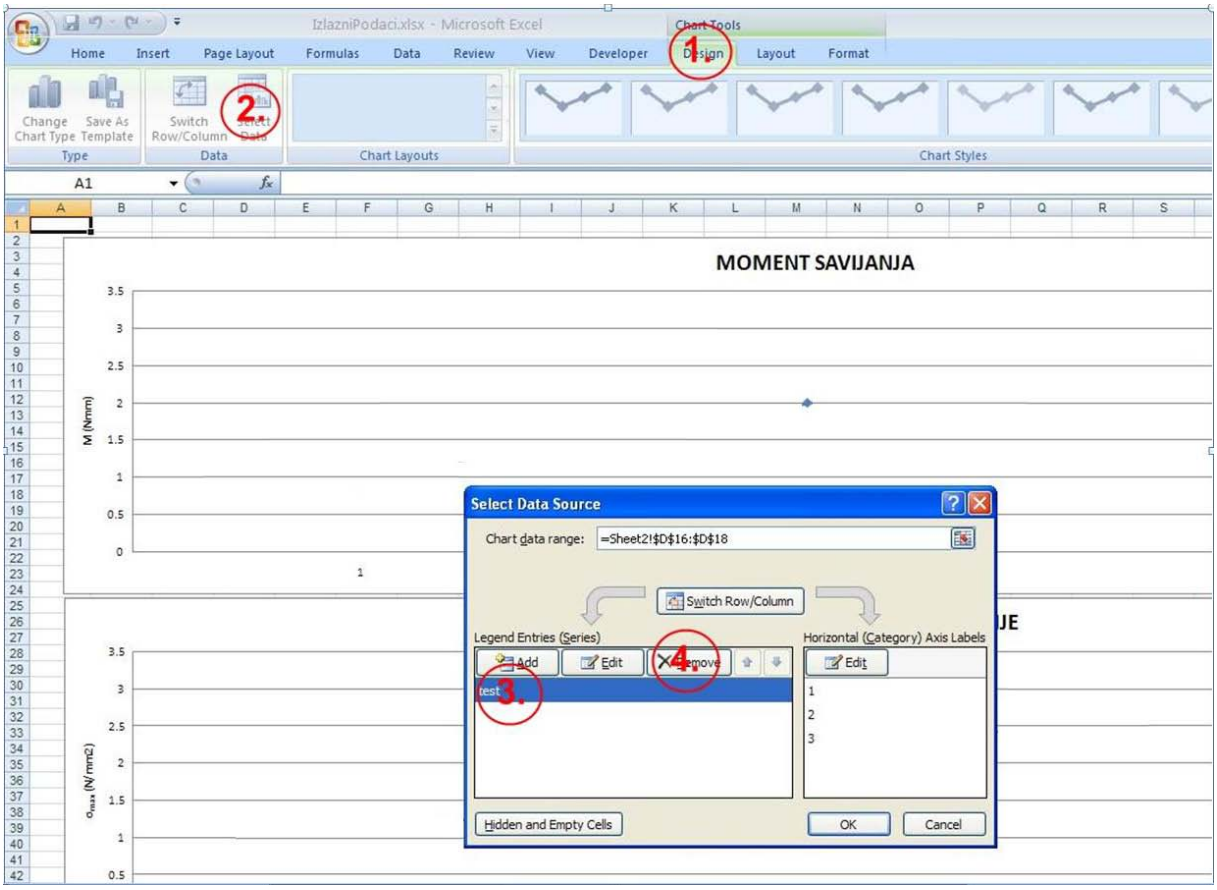
vratila. Naime vrijednosti iz polja „ca“ definiraju retke, u .xls datoteci, u koje se spremaju veličine naprezanja za određeno polje. Slikom [26] objašnjena je formula prema kojoj program svrstava izlazne podatke u stupac za dijagram. Slika služi samo kao shematski primjer izlaznih podataka i nije primjer stvarnog vratila, već segmenta prikazanog slikom [22] iz ranijeg primjera.



Slika 26: ***Shema izrade „dijagram“ stupca***

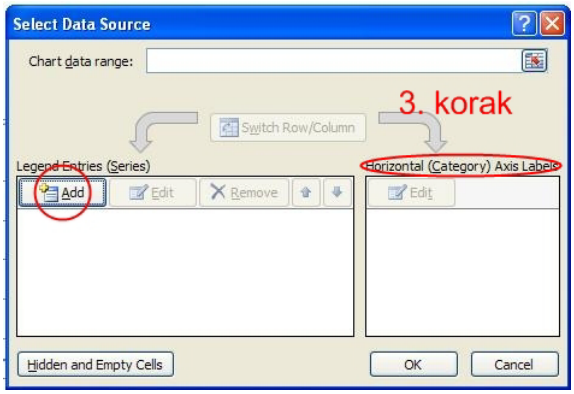
Greška aproksimacije koordinata polja povlači za sobom određenu grešku u rasporedu polja na dijagramima, no ona je manja od 0.5 mm, što je za naše potrebe zanemarivo. Prazna polja u stupcu “stupac za dijagram“ ne definiraju nikakvu vrijednost izlaznih podataka (*u našem slučaju moment savijanja*) za pripadajuće koordinate na apscisi, pa se zato te točke aproksimiraju.

Izrada dijagrama bazira se na predlošcima dijagrama u prvom radnom listu. Početni izgled dijagrama prikazan je slikom [24]. Unos podataka u dijagram prikazan je u nekoliko sljedećih koraka u kojima je prikazan unos momenta savijanja za dva proračunata vratila.

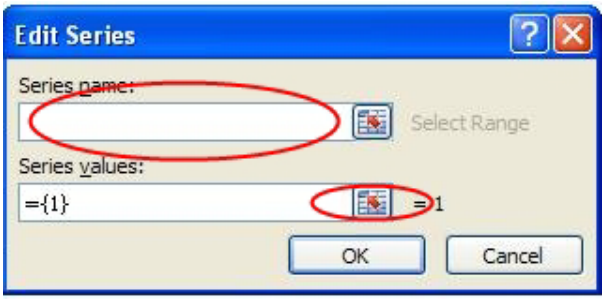


Slika 27: ***Brisanje „test“ krivulje***

- 1. Korak – brisanje „test“ krivulje [27]
  - 1. – „Design“ – otvaramo opcije za upravljanje dijagramom
  - 2. – „Select Data“ – sučelje za dodavanje i mijenjanje veličina u dijagramu
  - 3. – „test“ – odabiremo „test“ krivulju
  - 4. – „Remove“ – brišemo odabranu krivulju
- 2. Korak – dodavanje iznosa momenata
  - Pritiskom na tipku „Add“ [28] otvara se prozor [29]
  - U polje „Series name“ se unosi ime krivulje (npr. Vratilo 1, Vratilo 2, itd.)
  - U polje „Series values“ se odabire čitavi „Moment savijanja (dijagram)“ stupac u radnom listu željenog vratila [25].
  - Nakon potvrde pritiskom na tipku „OK“, dodaje se moment za drugo vratilo ponavljanjem drugog koraka

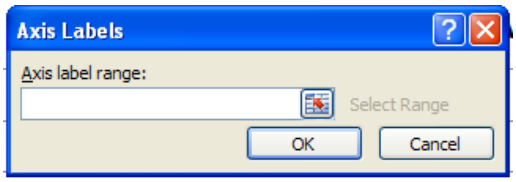


Slika 28: ***Dodavanje podataka za novo vratilo***



Slika 29: ***Odabir naziva i putanje***

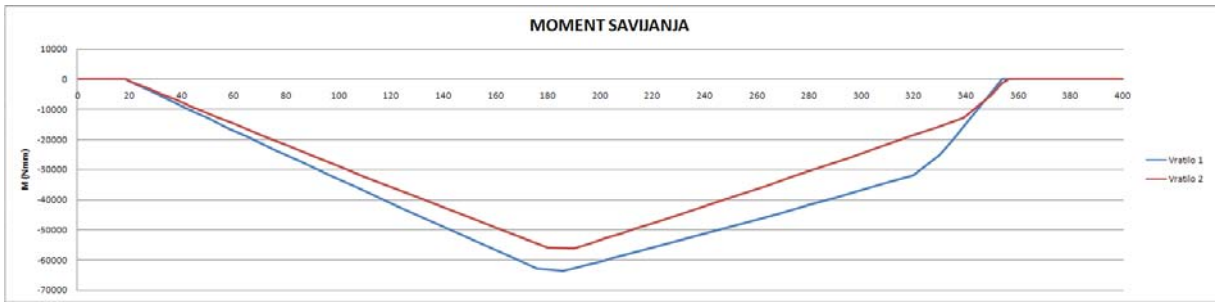
- 3. Korak – definicija osi apscisa
  - 1. – u polju „Horizontal (Category) Axis Label“ odabiremo tipku „Edit“ [28].
  - 2. – u polje „Axis label range „ koje se nalazi u novonastalom sučelju [30], unose se podaci iz stupca „x aksijalna os za dijagram“ od najdužeg vratila iz serije. {0,0.5,1,1.5,….,“duljina vratila“}



Slika 30: ***Odabir podataka za apscisu***

- 4. Korak – vizualno uređivanje dijagrama
  - Dijagram se može vizualno urediti u „Format“ sučelju dijagrama, koje se nalazi desno od „Design“ (1.) na slici [27].

Konačni izgled dijagrama za moment savijanja prikazan je slikom [31]. Na isti način u dijagrame se unose i ostali podaci o vratilu.



Slika 31: ***Konačni izgled dijagrama za 2 vratila***

7. ZAKLJUČAK

Program za proračun vratila metodom prijenosnih matrica ima zadaću skratiti vrijeme proračunavanja serije vratila i omogućiti usporedbu pojedinih vratila unutar serije. Kako bi se vratilo uopće moglo proračunati, potrebno je adekvatno pripremiti opis njegovih dimenzija te ostalih ulaznih podataka. Takva priprema također oduzima podosta vremena konstruktoru, posebno u razradi čitave serije. Stoga su automatizacija pripreme ulaznih podataka, kao i automatizacija ispisa, koje su također obuhvaćene u radu, vrlo vrijedne i korisne u analizi vratila.

Moguće je primijetiti da su procedure korištene u izradi programa poprilično kompleksne. Kompleksne su iz razloga što postoji velik broj podataka koji opisuju vratilo i koji su potrebni za pravilan proračun. Stoga je bilo podosta teško uspješno obuhvatiti sve ulazne podatke, a ujedno automatizirati što je moguće više rada u programu.

Korištenje programa zahtjeva određena znanja konstruktora i program ne može koristiti osoba koja se nikad nije susrela sa radom u CATIA-i ili nikad nije proračunala vratilo. Jasno je da sam program nije savršen i moguće ga je dosta doraditi, krenuvši od automatizacije izrade dijagrama. Isto je tako poznato da postoje određena odstupanja između točnih rezultata proračuna vratila i izlaznih podataka programa. Razlog je u tome što rezultati metode prijenosnih matrica ovise o broju polja na koje je vratilo podijeljeno. Kako god bilo rezultati su potpuno zadovoljavajući za potrebe usporedbe vratila, a sam program je više nego veliki korak u smjeru potpune automatizacije u razradi serije vratila.

Osnovna prednost programa opisanog u radu je, kao što je ranije napomenuto, velika ušteda konstruktorovog vremena. Time što smo ga oslobodili posla koji zahtjeva rutinski rad, a ne kreativnost, spriječili smo uzaludno trošenje tvrtkinog intelektualnog kapitala na opetovan, a ujedno standardan rad.

8. PRIMJERI

8.1. PRORAČUN SERIJE VRATILA PREDSTAVLJENIM PROGRAMOM

U svrhu boljeg razumijevanja rada sa programom za analizu vratila, predočen je primjer analize vratila te njihova usporedba. Za proračun su odabrana dva vratila iz serije vratila koju proizvodi tvrtka „Končar“.

Radionički crtež vratila jednak je primjeru predočenom u ranijem dijelu ovog rada [10]. Tablica sa iznosima kota kao i Design Table također su ranije prikazani slikama [12] i [11]. Iz tablice [12] kao primjer su odabrana vratila sa rednim brojem 9. i 19. Njihova razlika u dimenzijama vidljiva je na slici [32].

Postupak u programu za analizu vratila treba provesti za svako vratilo posebno zato jer vratila nisu susjedna na tablici pa Macro funkcija ne može napraviti zajedničku izlaznu datoteku. Usprkos tome rezultati analize se spremaju u zajedničku .xls datoteku te je time omogućena usporedba podataka. Slikama [33] i [34] prikazane su transformirane dimenzije vratila tj. dimenzije koje program koristi za proračun.

- vratila smo opteretili jednakim opterećenjem od:

  F_R= 20 kN – dodatna radijalna sila na osmom segmentu
  T= 610 Nm – torzioni moment tokom čitavog vratila

- odabrani tip ležaja je: „Kruti radijalni ležaj (kuglični)“
- položaj ležajeva: 5. i 11. Segment
- očekivana trajnost ležaja : 8000 h
- aksijalna sila na ležaj: 1250 N
- brzina vrtnje: 1420 min^{-1}

	Šifra		Tip			Materijal				A	A_1	B	B_1	C	C_1	ØD	$ØD_1$	$ØD_2$	$ØD_3$	$ØD_4$	$ØD_5$	E	E_1	G		
		Veličina	Duž	Pol	Ø	Sirova dužina	Broj dijeleva					>0.5	>0.5					K6	j6		K6					
9		225	Ma	6	90	1102	900 736			140	110	125	100	207	232	60	65	68	80	55	64.5	235	275	388		
19	11 005 160	280	S	8,10	110	1335	901 735			170	140	160	125	248	284	80	85	100	100	65	84.5	278	337	456.5		
	G_1	H	J	K	L	M	M_1			N	N_1	O	P	S	S_1	S_2	T	T_1	U	U_1	V	V_1	Z	Z_1	Z_2	
						6H	6H			0.1	0.1	0.1	H13	<0.2	<0.2		<0.2	<0.2	<0.2	<0.2	<0.2	<0.2	<0.2	<0.2		
9	301	635.5	238	10	1098	M20	M20			228	138.5	409	2.65	53	64	11	63.1	71.1	75.1	83.1	59.6	67.6	49	59	10	
19	364.5	779	344	10	1331	M20	M20			265.5	160.5	511	3.15	71	85	14	95.1	103.1	95.1	103.1	79.6	87.6	58	69	11	
	a	b	b_1	$Ød_1$	$Ød_2$	$Ød_3$	$Ød_4$	$Ød_5$	$Ød_6$	$Ød_7$	h_4	h	h_1	h_2	h_3	r	r_1	r_2	r_3	r_4	t	t_1	t_2			
				<0.2	h12	<0.1			<0.2	<0.2	>0.3	>0.3		>0.5							N9	N9	N9			
9	2	8	5	64.9	62	65.3	89	76	79.9	64.4	100	100	3	230	10	1.6	9	8	6	6	18	16	12		
19	2	5	8	84.9	81.5	85.3	109	96	99.9	84.4	100	100	4	334	10	2	11	9	6	6	22	18	12		

Slika 32: ***Podaci o 9. I 19. Vratilu***

Broj segmenta	Dužina segmenta:		Promjer segmenta:		Radijus zaobljenja:	
1	A	140	ØD	60	r	1.6
2	G-N-A	20	$Ød_1$	64.9		1
3	P	2.65	$Ød_2$	62	1	1
4	N-(G-C)-P	44.35	$ØD_1$	65	r	1.6
5	E-C	28	$Ød_3$	65.3	r	1.6
6	G-E	153	$ØD_2$	68	r	1.6
7	L-H-G	74.5	$Ød_4$	89		0
8	J-K	228	$ØD_3$	80	1	1
9	K	10	$Ød_5$	76	1	1
10	H-J-G_1	96.5	$Ød_6$	79.9		0
11	G_1-E_1	26	$Ød_3$	65.3	r	1.6
12	E_1-C_1	43	$ØD_1$	65	1	1
13	N_1-(G_1-C_1)-P	66.85	$ØD_5$	64.5	1	1
14	P	2.65	$Ød_2$	62	1	1
15	G_1-N_1-A_1	52.5	$Ød_7$	64.4		1
16	A_1	140	$ØD_4$	55	r	1.6

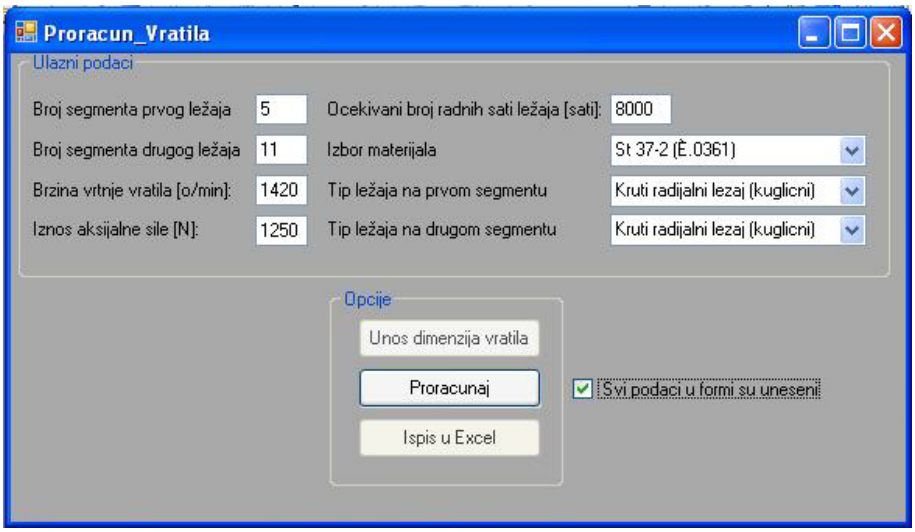
Slika 33: ***Transformirane kote za 9. vratilo***

Broj segmenta	Dužina segmenta:		Promjer segmenta:		Radijus zaobljenja:	
1	A	170	ØD	80	r	2
2	G-N-A	21	$Ød_1$	84.9		1
3	P	3.15	$Ød_2$	81.5	1	1
4	N-(G-C)-P	53.85	$ØD_1$	85	r	2
5	E-C	30	$Ød_3$	85.3	r	2
6	G-E	178.5	$ØD_2$	100	r	2
7	L-H-G	95.5	$Ød_4$	109		0
8	J-K	334	$ØD_3$	100	1	1
9	K	10	$Ød_5$	96	1	1
10	H-J-G_1	70.5	$Ød_6$	99.9		0
11	G_1-E_1	27.5	$Ød_3$	85.3	r	2
12	E_1-C_1	53	$ØD_1$	85	1	1
13	N_1-(G_1-C_1)-P	76.85	$ØD_5$	84.5	1	1
14	P	3.15	$Ød_2$	81.5	1	1
15	G_1-N_1-A_1	64	$Ød_7$	84.4		1
16	A_1	170	$ØD_4$	65	r	2

Slika 34: ***Transformirane kote za 19. vratilo***

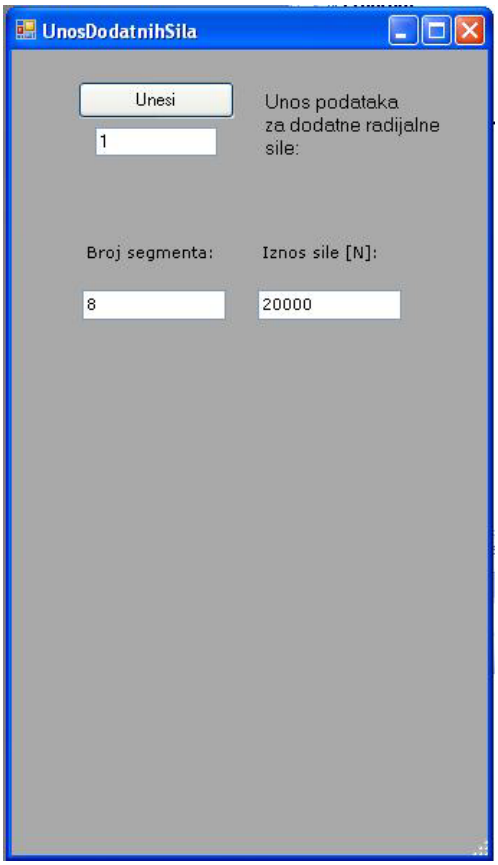
Postupak u koracima:

1. Odabir vratila 9. u tablici [12] i pokretanje Macro opcije „Start“. Spremanje i zatvaranje .xls datoteke sa odgovarajućom tablicom.
2. Pokretanje programa za proračun.
3. Odabir tipke „Unos dimenzija“ te odabir gore pripremljene .xls datoteke.



Slika 35: ***Ulazni podaci***

4. Unos podataka u polja „Ulazni podaci“.
5. Odabir „Okvira za izbor“ te pokretanje tipke „Proračunaj“.
6. Unos dodatnih sila u odgovarajuće sučelje [36].
7. Unos torzionog momenta [37].

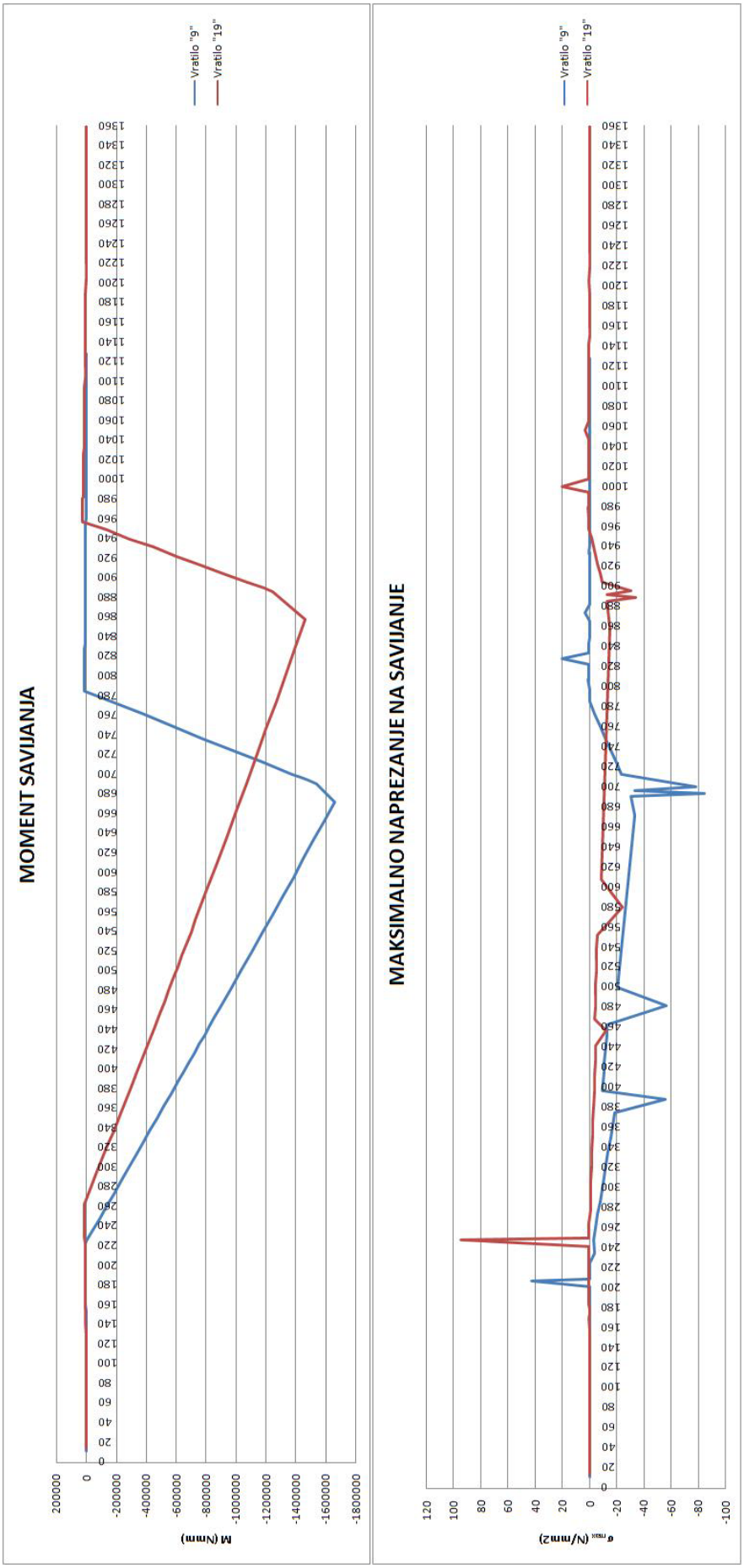


Slika 36: ***Dodatno radijalno opterećenje***

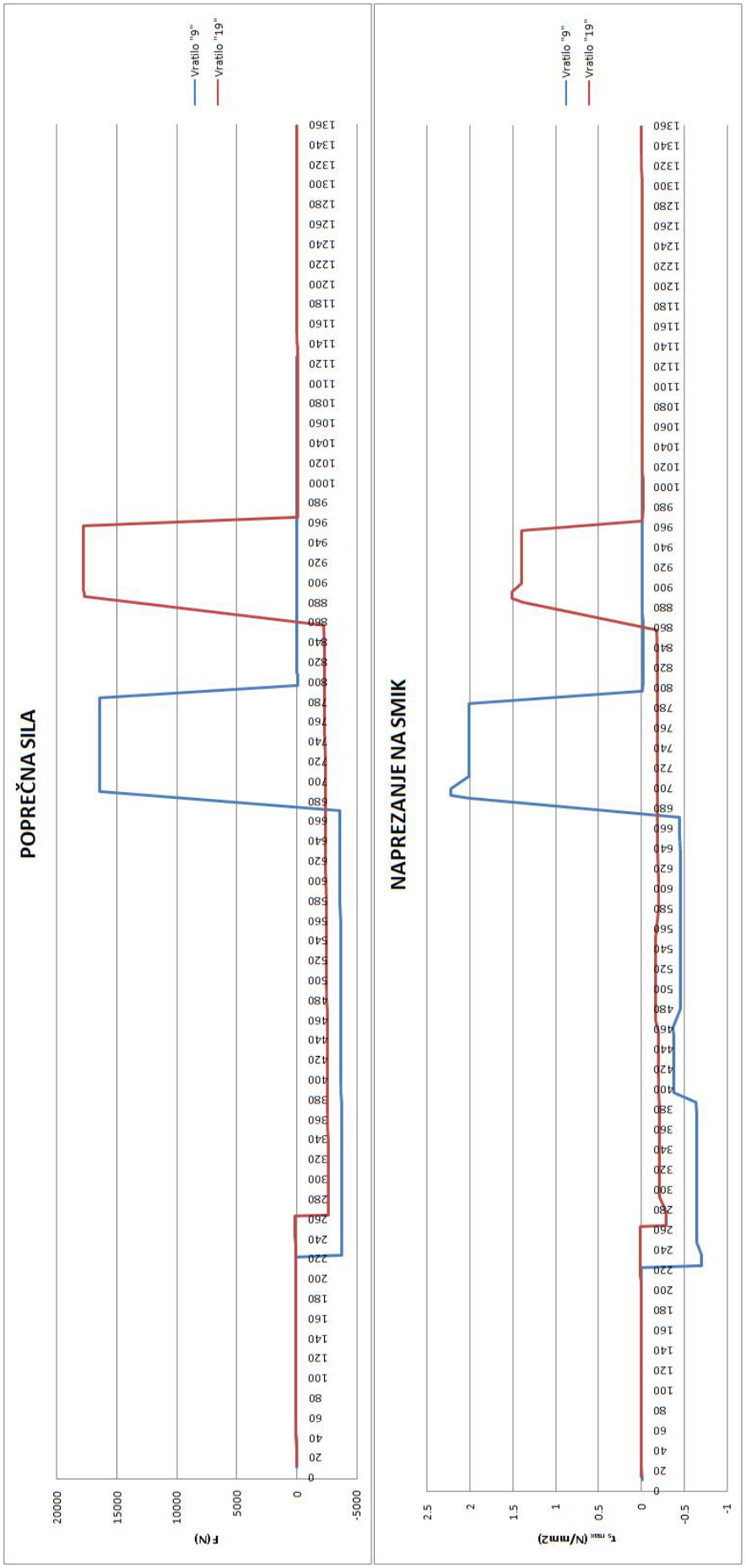
Slika 37: ***Torzioni moment***

8. Odabir ležaja koji služi kao putanja na mapu.
9. Pokretanje opcije „Ispis u Excel“ i odabir datoteke u koju se spremaju podaci.
10. Zatvaranje osnovnog sučelja u programu. (*automatski se gasi Excel*)
11. Čitav postupak treba ponoviti za sljedeće vratilo.

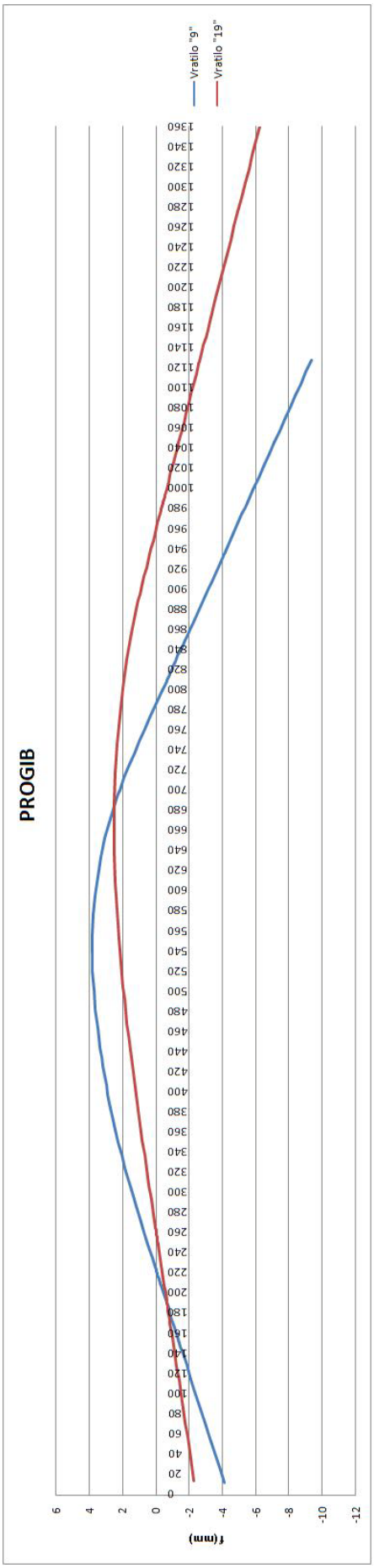
Nakon provedenog proračuna izlazna datoteka sadrži dva radna lista prikazana slikom [25]. Svaki od njih sadrži podatke o jednom od analiziranih vratila. U radnom listu „Dijagrami“ se nalazi predlošci dijagrama u koje treba unijeti izlazne podatke kako bi mogli usporediti. Dijagrami izlaznih veličina za primjerom analizirana vratila prikazani su sljedećim slikama.



Slika 38: ***Dijagram momenta savijanja i maksimalnog naprezanja na savijanje***



Slika 39: ***Dijagram poprečne sile i naprezanja na smik***



Slika 40: ***Dijagram progiba***

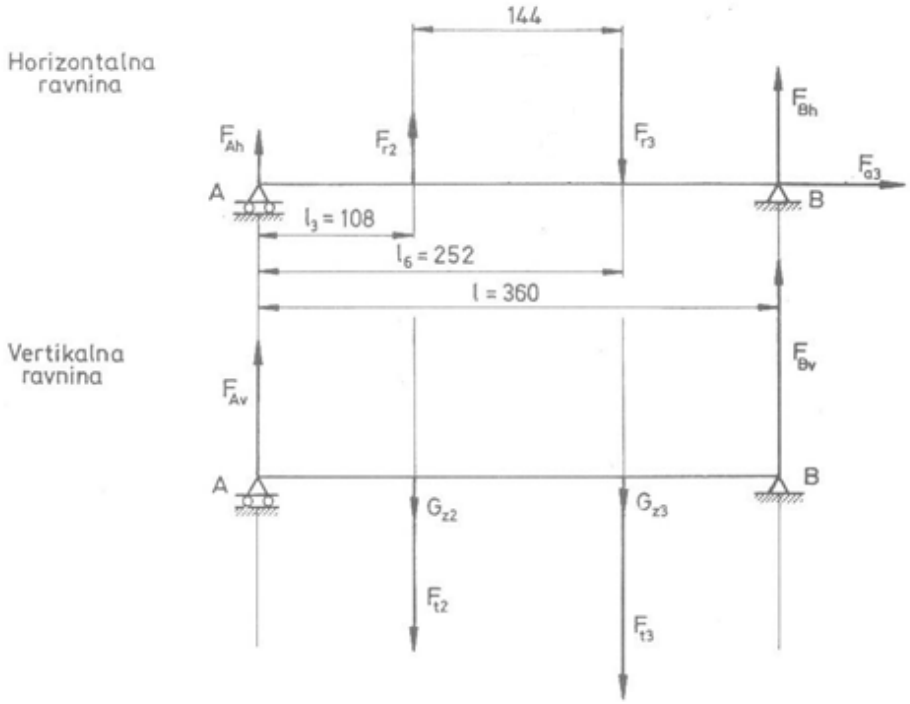
8.2. USPOREDBA REZULTATA ANALITIČKI TE PROGRAMOM PRORAČUNATOG VRATILA

Za usporedbu rezultata je odabrano vratilo iz predloška za proračun vratila [6]. Analitičkim putem je proračunat maksimalni progib vratila te je rezultat uspoređen sa dijagramom dobivenim iz programa. Napomenimo još samo kako aksijalne sile nisu uračunate u proračunu jer procedura za proračun metodom prijenosnih matrica također ne uzima u obzir aksijalne sile.

8.2.1. ANALITIČKO RJEŠENJE

Prema lit. [6] opterećenja na vratilo su sljedeća:

$F_{t_2} = 3389$ N
$F_{r_2} = 1233$ N
$F_{t_3} = 9667$ N
$F_{r_3} = 3700$ N



Slika 41: ***Opterećenja na vratilo***

Komponente radijalnih reakcija u A i B:

$$\Sigma M_{B_h} = 0 \ \therefore \ F_{A_h} \cdot l + F_{r_2}(l - l_3) - F_{r_3}(l - l_6) = 0$$

$$F_{A_h} = \frac{F_{r_3}(l - l_6) - F_{r_2}(l - l_3)}{l} = \frac{3700(360 - 252) - 1233(360 - 108)}{360} = 247 \text{ N}$$

$$\Sigma Y_h = 0 \ \therefore \ F_{B_h} = -F_{A_h} - F_{r_2} + F_{r_3} = -247 - 1233 + 3700 = 2220 \text{ N}$$

$$\Sigma M_{B_v} = 0 \ \therefore \ F_{A_v} \cdot l - (F_{t_2} + G_{z_2})(l - l_3) - (F_{t_3} + G_{z_3})(l - l_6) = 0$$

$$F_{A_v} = \frac{(F_{t_2} + G_{z_2})(l - l_3) + (F_{t_3} + G_{z_3})(l - l_6)}{l} =$$

$$= \frac{(3389 + 240) \cdot 252 + (9667 + 110) \cdot 108}{360} = 5473 \text{ N}$$

$$\Sigma Y_v = 0 \ \therefore \ F_{B_v} = F_{t_2} + G_{z_2} + F_{t_3} + G_{z_3} - F_{A_v} = 3389 + 240 + 9667 + 110 - 5473$$

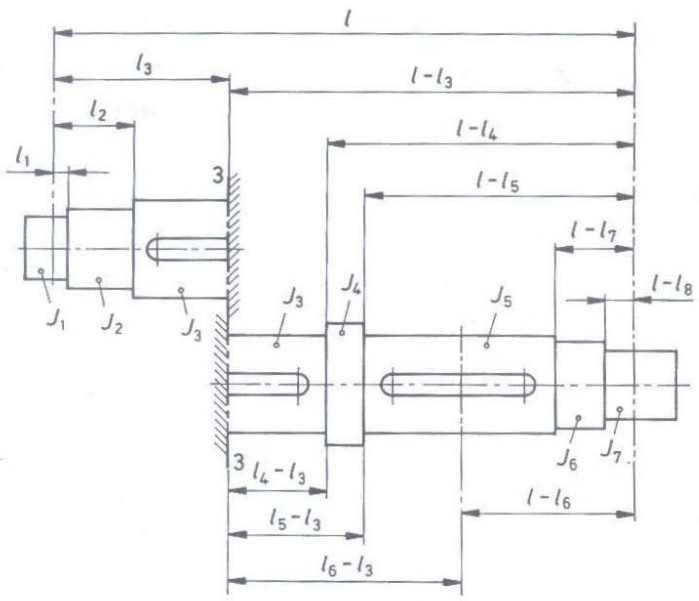
$$= 7933 \text{ N}$$

Rezultante radijalne reakcije u A i B:

$$F_A = \sqrt{{F_{A_h}}^2 + {F_{A_v}}^2} = \sqrt{247^2 + 5473^2} = 5479 \text{ N}$$

$$F_B = \sqrt{{F_{B_h}}^2 + {F_{B_v}}^2} = \sqrt{2220^2 + 7933^2} = 8238 \text{ N}$$

Sljedećim računom proračunavamo savojne elastične deformacije vratila. Račun se provodi metodom zamišljenog uklještenja na mjestu djelovanja aktivnih sila. Kako je vratilo opterećeno na mjestima 3 i 6, na ista stavljamo fiktivna uklještenja. Prema tome, ravnajući se prema lit. [6], prvo ukliještimo mjesto 3:



J[mm⁴]=moment tromosti presjeka

Za puni kružni presjek

$J = \frac{\pi d^4}{64} \cong 0{,}05\,d^4$

$E\left[\text{N/mm}^2\right]$=modul elastičnosti materijala vratila

Za čelik:
E=200000 N/mm²

Slika 42: ***Uklještenje na mjestu 3.***

Parcijalne horizontalne komponente radijalnih reakcija u osloncima uslijed horizontalne sile F_{r_2} u presjeku 3:

$$F_{A_{h3}} = \frac{F_{r_2}(l - l_3)}{l} = \frac{1233(252)}{360} = 863 \text{ N}$$

$$F_{B_{h3}} = F_{r_2} - F_{A_{h3}} = 1233 - 863 = 370 \text{ N}$$

Progib konzole u osloncu A:

$$f_{A_{h3}} = 6{,}8 \cdot \frac{F_{A_{h3}}}{E}\left(\frac{l_1^3}{d_1^4} + \frac{l_2^3 - l_1^3}{d_2^4} + \frac{l_3^3 - l_2^3}{d_3^4}\right) = 6{,}8 \cdot \frac{863}{200000}\left(\frac{9^3}{40^4} + \frac{48^3 - 9^3}{49^4} + \frac{108^3 - 48^3}{60^4}\right) = 0{,}00317\text{mm}$$

Progib konzole u osloncu B:

$$f_{\mathrm{B_{h3}}} = 6{,}8 \cdot \frac{F_{\mathrm{B_{h3}}}}{E}\left(\frac{(l-l_8)^3}{d_7^4} + \frac{(l-l_7)^3-(l-l_8)^3}{d_6^4} + \frac{(l-l_5)^3-(l-l_7)^3}{d_5^4} + \frac{(l-l_4)^3-(l-l_5)^3}{d_4^4} + \frac{(l-l_3)^3-(l-l_4)^3}{d_3^4}\right) = 0{,}01434 \text{ mm}$$

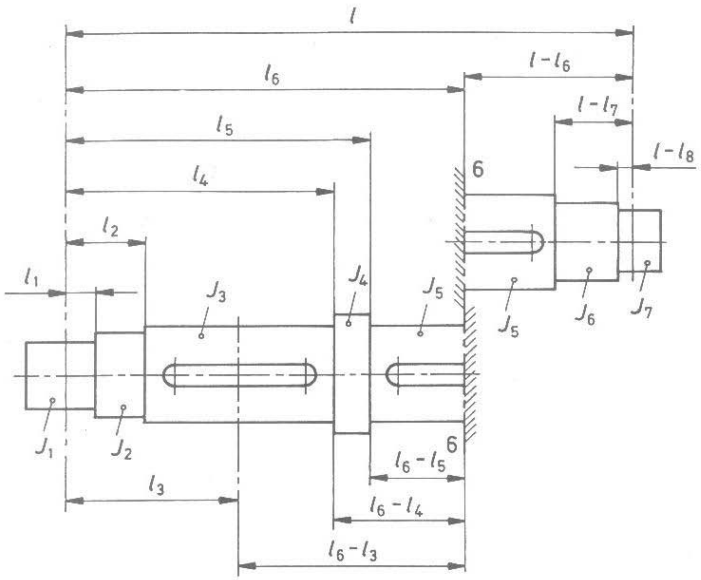
Progib ispod sile F_{r_2} - na mjestu 3:

$$f'_{\mathrm{h_3}} = f_{\mathrm{A_{h3}}} + \frac{(f_{\mathrm{B_{h3}}} - f_{\mathrm{A_{h3}}}) \cdot l_3}{l} = 0{,}00317 + \frac{(0{,}01434 - 0{,}00317) \cdot 108}{360} = 0{,}006521 \text{ mm}$$

Preostalo nam je još prema istom postupku proračunat parcijalne progibe za vertikalnu ravninu u slučaju kad je ukliještena točka 3 te za vertikalni i horizontalni slučaj kad je ukliještena točka 6. Ovaj dio proračuna je razrađen rukom, no kako je identičan gornjem nije upisan u rad. Navedena su samo rješenja.

Vertikalna ravnina na mjestu 3:

$$f'_{v_3} = 0{,}01919 \text{ mm}$$



U presjeku 6 djeluju u horizontalnoj ravnini sile F_{r_3} i F_{a_3}, a u vertikalnoj ravnini sile F_{t_3} i G_{z_3}. Postupak određivanja parametara parcijalnih elastičnih linija u horizontalnoj i vertikalnoj ravnini uslijed djelovanja sila u hvatištu 6 identičan je onome primijenjenom kod uklještenja vratila na mjestu 3.

Slika 43: ***Uklještenje na mjestu 6.***

Horizontalna ravnina na mjestu 6:

$$f'_{\mathrm{h_6}} = 0{,}02877 \text{ mm}$$

Vertikalna ravnina na mjestu 6:

$$f'_{\mathrm{v_6}} = 0{,}07596 \text{ mm}$$

Ukupni horizontalni i vertikalni progib:

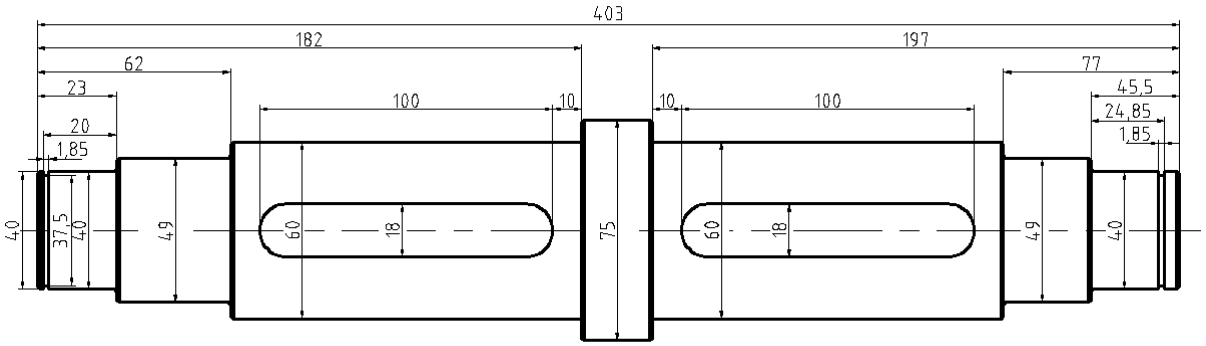
$$f_{\mathrm{h}} \cong f'_{\mathrm{h3}} + f'_{\mathrm{h6}} = 0{,}006521 + 0{,}02877 = 0{,}035291 \text{ mm}$$
$$f_{\mathrm{v}} \cong f'_{\mathrm{v3}} + f'_{\mathrm{v6}} = 0{,}01919 + 0{,}07596 = 0{,}09515 \text{ mm}$$

Ukupni ili maksimalni rezultirajući progib:

$$f = \sqrt{f_{\mathrm{h}}^2 + f_{\mathrm{v}}^2} = \sqrt{0{,}035291^2 + 0{,}09515^2} = 0{,}10148 \text{ mm}$$

8.2.2. PRORAČUN PROGRAMOM U VISUAL BASICU

Vratilo istih dimenzija i opterećenja analizirano je programom opisanim u radu.



Slika 44: ***Dimenzije analiziranog vratila***

Opterećenja:

- iznos sile na petom segmentu (položaj prvog pera) prema lit. [6]:

$$F_5 = \sqrt{{F_{\mathrm{r}_2}}^2 + (F_{\mathrm{t}_2} + G_{\mathrm{z}_2})^2} = \sqrt{1233^2 + (3389 + 240)^2} = 3832 \text{ N}$$

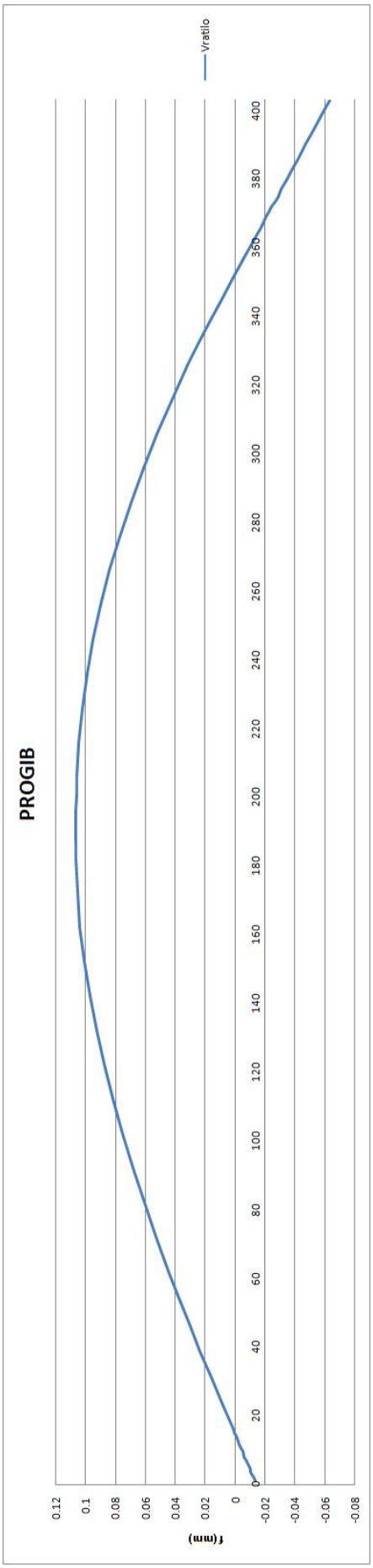
- iznos sile na sedmom segmentu (položaj drugog pera) prema lit. [6]:

$$F_7 = \sqrt{{F_{\mathrm{r}_3}}^2 + (F_{\mathrm{t}_3} + G_{\mathrm{z}_3})^2} = \sqrt{3700^2 + (9667 + 110)^2} = 10453 \text{ N}$$

- torzioni moment 610 000 Nmm
  odabrani tip ležaja je: „Kruti radijalni ležaj (kuglični)“
  položaj ležajeva: 3. i 9. Segment
  očekivana trajnost ležaja : 8000 h
  brzina vrtnje: 1420 min^{-1}

Na temelju gore navedenih podataka dobiven je dijagram progiba prikazan slikom [45]. iz dijagrama je vidljivo da je maksimalni progib dobiven analizom jednak iznosu $f = \ 0.106572\ mm$, što je približno jednako analitičkom rješenju.

PROGIB	
Analitički	$f = \ 0.10148\ mm$
Programom	$f = \ 0.106572\ mm$



Slika 45: ***Progib vratila***

9. LITERATURA

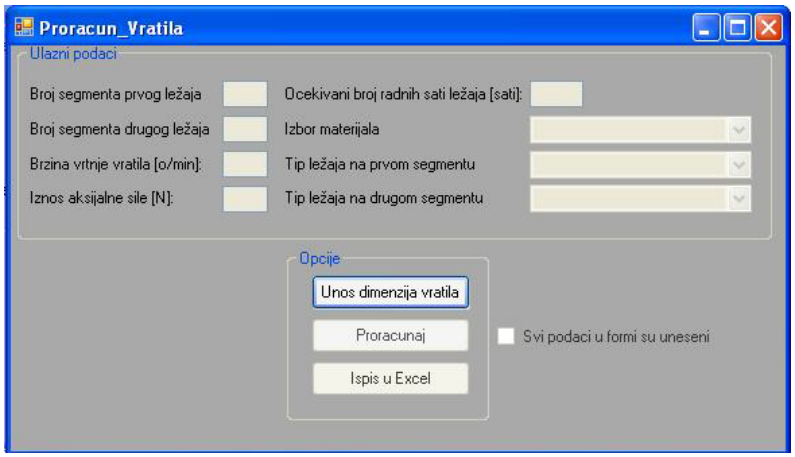
[1] http://hr.wikipedia.org/wiki/Microsoft_Excel
[2] http://www.pcchip.hr/vijesti/recenzije-softvera/catia-v5-r11
[3] www.graphicon.ru
[4] www.edmpdm.de
[5] Kraut B., Strojarski priručnik, Tehnička knjiga, Zagreb, 1986.
[6] Horvat Z., Vratilo (Proračun), Fakultet strojarstva i brodogradnje, Zagreb
[7] http://social.msdn.microsoft.com/Forums/en/vbinterop/thread/a98b7675-c2d5-4036-bbde-53a3b88a4df5
[8] Pavković N., Programi za proračun vratila i ležaja te crtanje vratila, Končar, Zagreb, 1991.
[9] http://www.homeandlearn.co.uk/NET/vbNET.html

10. DODATAK

DODATAK

(Programski kod upotpunjen komentarima)

Osnovno sučelje programa za proračun vratila



```
Public Class Proracun_Vratila
    'U pocetku programa bilo je potrebno definirati globalne varijable koje
ce se koristiti u daljnjem programiranju.
    'Razlika izmedju globalni i lokalnih varijabli je u cinjenici da
globalne mogu koristiti sve funkcije unutar projekta, a lokalne samo
funkcije u kojima se one nalaze
    'Kao što je vidljivo iz priloženog svaka varijabla je odredjenog tipa,
koji je definira i daje joj funkciju.
    Dim W(500, 50), P(500, 50), OM(500, 50), Q(500, 50), p_s(500, 50),
Lhl(50), Lhd(50), Pl(50), Pd(50) As Double
    Dim f_l(100), sila_l(100), f_d(100), sila_d(100), d_s(1000, 50),
rad(100, 50), stsila_l(100), stsila_d(100) As Double
    Dim masa(500, 50), mi(500, 50), pro(500, 50), pr_seg(500, 50),
fdod(500, 500), seg(500, 50), Lk(500, 50), Lk2(500, 50), smax(500, 50),
tmax(500, 50), ttmax(500, 50) As Double
    'pro(500, 50) - 2D polje koje sadrzi iznose promjera segmenta
cijelokupne serije vratila. Prva dimenzija oznacava indeks segmenta, a
druga ideks vratila u seriji.
    'd_s(1000, 50) - duljine segmenata
    'rad(100, 50) - prijelazni radijusi segmenata
    'W(500, 50) - 2D polje koje sadrzi iznose progiba polja cijelokupne
serije vratila. Prva dimenzija oznacava indeks polja, a druga ideks vratila
u seriji.
    'P(500, 50) - nagibi polja
    'OM(500, 50) - momenti savijanja na polju
    'Q(500, 50) - poprecne sile na polju
    'p_s(500, 50) - sirine polja
    'pr_seg(500, 50) - promjeri polja
    'masa(500, 50) - mase polja
    'mi(500, 50) - momenti otpora kružnih presjeka polja
    'f_l(100) - polje s sirinama prvog lezaja
    'f_d(100) - polje s sirinama drugog lezaja
    'sila_l(100) - polje sa dozvoljenim dinamickim naprezanjima za prvi
lezaj
    'sila_d(100) - polje sa dozvoljenim dinamickim naprezanjima za drugi
lezaj
    'stsila_l(100) - polje sa dozvoljenim statickim naprezanjima za prvi
lezaj
    'stsila_d(100) - polje sa dozvoljenim statickim naprezanjima za drugi
lezaj
    'fdod(500, 500) - iznosi dodatnih sila kojima su opterecena polja
vratila
    'seg(500, 50) - varijabla oznacava polje koje sadrzi informacije o tome
kojem segmentu pripada odredjeno polje vratila
    'Lk(500, 50) - polje sa faktorima zareznog djelovanja kod savijanja
    'Lk2(500, 50) - polje sa faktorima zareznog djelovanja kod uvijanja
    'smax(500, 50) - polje sa iznosima maksimalnog naprezanja na savijanje
za sva polja vratila
    'tmax(500, 50) - polje sa iznosima maksimalnog naprezanja na smik za
sva polja vratila
    'ttmax(500, 50) - polje sa iznosima maksimalnog naprezanja na uvijanje
za sva polja vratila
    'Lhl(50) - ocekivani broj radnih sati prvog lezaja za seriju vratila
    'Lhd(50) - ocekivani broj radnih sati drugog lezaja za seriju vratila
    'Pl(50) - dinamicka opterecenost prvih lezaja u seriji vratila
    'Pd(50) - dinamicka opterecenost drugih lezaja u seriji vratila
    Dim nr(100), p11(50), length(50), di(50) As Integer
    'nr(100) - brojac koji definira broj segmenata odredjenog vratila u
seriji
    'p11(50) - brojac koji definira broj polja odredjenog vratila u seriji
    'length(50) - ukupne duzine svih vratila u seriji
```

```
    'di(50) - brojac koji definira broj koordinata na apscisi dijagrama za
sva vratila u seriji
    Dim p1, z, l1(50), l2(50), j, k, mm, n, dr, i As Integer
    Dim bl1, bl2, Gdn, Gfdn, Tdi, s_s, ss, bll, bll2, nm As Double
    'l1(50), l2(50) - redni brojevi polja na kojima se nalaze lezajevi
    'mm - broj polja vratila u kodu prijenosnih matrica
    'bl1 - indeks segmenta nakojemu se nalazi prvi lezaj
    'bl2 - indeks segmenta nakojemu se nalazi drugi lezaj
    'Gdn - dopusteno naprezanje na vlak/tlak za odabrani materijal
    'Gfdn - dopusteno naprezanje na savijanje za odabrani materijal
    'Tdi - dopusteno naprezanje na uvijanje za odabrani materijal
    's_s - sirina ostalih polja (polja na kojima se ne nalazi lezaj) na
segmentu sa prvim lezajom
    'ss - sirina ostalih polja (polja na kojima se ne nalazi lezaj) na
segmentu sa drugim lezajom
    'bll - razlika izmedju duljine segmenta i sirine lezaja koji se nalazi
na njemu - za prvi lezaj
    'bll2 - razlika izmedju duljine segmenta i sirine lezaja koji se nalazi
na njemu - za drugi lezaj
    'nm - brzina vrtnje vratila
    Dim bool(500, 50) As Boolean 'varijabla koja definira polja koja su
opterecena torzionim momentom
    Dim path2 As String 'adresa mape sa .txt datotekama koje opisu lezaje
    Dim UnosDodatnihSila As New UnosDodatnihSila 'definira formu
UnosDodatnihSila
    Dim UnosTorMomenta As New UnosTorMomenta 'definira formu UnosTorMomenta
    'Varijable u bloku proba su preuzete iz gotovog koda koji smo dobili od
Dr. sc. N. Pavkoviæa.
    'Njihovo objašnjenje nalazi se u dobivenoj dokumentaciji.
    'Proba
    Dim V(1000), U(1000), T(1000), S(1000) As Double
    Dim R(1000), AK(1000), Y(1000) As Double
    Dim H(1000), F(1000), D(1000), B(1000) As Double
    Dim A(1000), A1(1000), A4(1000) As Double
    Dim A3(1000), D1(1000), F1(1000), F3(1000) As Double
    Dim Y1(1000), AK1(1000), S1(1000) As Double
    Dim C(1000), SILA(1000), CRU(5) As Double
    Dim EY As Double
    Dim JT As Integer
    Dim TJ(1000), VR(1000), PROM(1000), GM(1000), EL(1000) As Double
    'bilo je u common bloku
    Dim TT, NP(5), IBLAS, ZL10H(5) As Integer
    'bilo je u common bloku
    'COMMON /A/ TJ(MAX),VR(MAX),PROM(MAX), */A2/ GM(MAX),EL(MAX), */B2/
MM,TT,JX,NP(MT),IBLAS,ZL10H(MT)
    Dim AKO, ALI, WO, PO As Double
    Dim jx As String
    Dim mm1, jx1 As Integer
    'Kraj Proba
    'Naredna funkcija služi za izvlacenje podataka (dimenzija vratila) iz
excel datoteke.
    'Podaci su prvotno ispisani u Datagridview te se kasnije prebacuje u
Listbox-ove.
    Private Sub BtnUnosdimenzija_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles BtnUnosdimenzija.Click
        'sljedecih nekoliko redaka sluzi za ostvarivanje veze sa Miscrsoft
Excelom te kopiranje podataka iz odabranog sheet-a u datagridview
        'takodjer su definiranje potrebne varijable za stvaranje veze
        Try
            Dim MyConnection As System.Data.OleDb.OleDbConnection
            Dim DtSet As System.Data.DataSet
```

```
            Dim MyCommand As System.Data.OleDb.OleDbDataAdapter
            Dim oExcel As Microsoft.Office.Interop.Excel.Application = New
Microsoft.Office.Interop.Excel.Application
            Dim fileD As New OpenFileDialog
            fileD.ShowDialog() 'otvara OpenFileDialog
            Dim fileName As String = fileD.FileName
            MyConnection = New
System.Data.OleDb.OleDbConnection("provider=Microsoft.Jet.OLEDB.4.0;Data
Source='" & fileName & "';Extended Properties=Excel 8.0;")
            MyCommand = New System.Data.OleDb.OleDbDataAdapter("select *
from [Sheet3$]", MyConnection)
            MyCommand.TableMappings.Add("Table", "TestTable")
            DtSet = New System.Data.DataSet
            MyCommand.Fill(DtSet)
            Datagridview.DgwLista.DataSource = DtSet.Tables(0)
            MyConnection.Close()
            Datagridview.ShowDialog() 'otvara datagridview sa unesenim
dimenzijama
            If Datagridview.DialogResult =
Windows.Forms.DialogResult.Cancel Then
                Me.Close() 'u slucaju da je na formi datagidview odabrana
opcija "Cancel" program se prekida
            End If
            Dim i, j, n1, n2 As Integer 'lokalni brojaci
            For j = 0 To Datagridview.DgwLista.ColumnCount - 1 'petlja koja
odredjuje stupac u datagridview-u na koji se odnose zadane naredbe
                n1 = 0
                n2 = 0
                For i = 0 To (Datagridview.DgwLista.RowCount / 3) - 1
'petlja koja odredjuje redak u datagridview-u na koji se odnose zadane
naredbe
                    nr(j) = nr(j) + 1
                    d_s(i, j) = CDbl(Datagridview.DgwLista.Item(j,
i).Value()) 'Sprema vrijednost koja se nalazi u petljama definiranom polju.
U ovom slucaju to je duljina segmenta.
                    length(j) = length(j) + d_s(i, j)
                Next i
                For i = Datagridview.DgwLista.RowCount / 3 To 2 / 3 *
Datagridview.DgwLista.RowCount - 1
                    pro(n1, j) = CDbl(Datagridview.DgwLista.Item(j,
i).Value) 'sprema promjer segmenta
                    n1 = n1 + 1
                Next i
                For i = 2 / 3 * Datagridview.DgwLista.RowCount To
Datagridview.DgwLista.RowCount - 1
                    rad(n2, j) = CDbl(Datagridview.DgwLista.Item(j,
i).Value) 'sprema zaobljenje na rubu segmenta
                    n2 = n2 + 1
                Next i
            Next j
            'dolje navedene naredbe manipuliraju aktivnoscu odredjenih
kontrola na formi
            CheckBox1.Enabled = True
            txt_Fa.Enabled = True
            TxtBrojradnihsati.Enabled = True
            TbBrSegDrugog.Enabled = True
            TbBrSegPrvog.Enabled = True
            TbBrzinaVrtnje.Enabled = True
            CbIzborMaterijala.Enabled = True
            CbTipLezajaDrugi.Enabled = True
            CbTipLezajaPrvi.Enabled = True
```

```
            BtnUnosdimenzija.Enabled = False
        Catch ex As Exception
            MessageBox.Show("Došlo je do greške u unosu.")
            Me.Close()
        End Try
    End Sub
    Private Sub ObradaIzboraMaterijala()
        'funkcija definira podatke o materijalu na temelju materijala
odabranog u combobox-u
        If CbIzborMaterijala.Text = "St 37-2 (C.0361)" Then
            Gdn = 170
            Gfdn = 190
            Tdi = 140
        ElseIf CbIzborMaterijala.Text = "St 42-2 (C.0461)" Then
            Gdn = 200
            Gfdn = 210
            Tdi = 160
        ElseIf CbIzborMaterijala.Text = "St 52-3 (C.0561)" Then
            Gdn = 220
            Gfdn = 240
            Tdi = 190
        ElseIf CbIzborMaterijala.Text = "St 60-2 (C.0645)" Then
            Gdn = 280
            Gfdn = 300
            Tdi = 230
        ElseIf CbIzborMaterijala.Text = "St 70-2 (C.0745)" Then
            Gdn = 330
            Gfdn = 350
            Tdi = 260
        End If
    End Sub
    Private Sub odabirlezaja()
        'Funkcija kojom biramo segmenate na koje smještamo ležajeve i tip
ležaja
        'na temelju odabranih velicina funkcija sama ucitava podatke o
lezaju
        Dim i As Integer
        Dim doc1, doc2, docfull As String 'varijable koje definiraju
djelove adrese za potrebne lezajeve
        bl1 = CInt(TbBrSegPrvog.Text) - 1
        bl2 = CInt(TbBrSegDrugog.Text) - 1
        'petlja koja definira doc1 varijablu na temelju odabranog lezaja u
comboboxu
        If CbTipLezajaPrvi.Text = "Kruti radijalni lezaj (kuglicni)" Then
            doc1 = "kl"
        ElseIf CbTipLezajaPrvi.Text = "Bacvasto valjkasti lezaj" Then
            doc1 = "bv"
        ElseIf CbTipLezajaPrvi.Text = "Cilindricno valjkasti lezaj" Then
            doc1 = "cv"
        ElseIf CbTipLezajaPrvi.Text = "Iglicasti lezaj" Then
            doc1 = "il"
        ElseIf CbTipLezajaPrvi.Text = "Aksijalni kuglicni lezaj" Then
            doc1 = "akl"
        ElseIf CbTipLezajaPrvi.Text = "Aksijalni valjkasti lezaj" Then
            doc1 = "avl"
        End If
        If CbTipLezajaDrugi.Text = "Kruti radijalni lezaj (kuglicni)" Then
            doc2 = "kl"
        ElseIf CbTipLezajaDrugi.Text = "Bacvasto valjkasti lezaj" Then
            doc2 = "bv"
        ElseIf CbTipLezajaPrvi.Text = "Cilindricno valjkasti lezaj" Then
```

```
            doc2 = "cv"
        ElseIf CbTipLezajaPrvi.Text = "Iglicasti lezaj" Then
            doc2 = "il"
        ElseIf CbTipLezajaPrvi.Text = "Aksijalni kuglicni lezaj" Then
            doc2 = "akl"
        ElseIf CbTipLezajaPrvi.Text = "Aksijalni valjkasti lezaj" Then
            doc2 = "avl"
        End If
        Try
            'adresa lezaja "docfull" sastoji se od adrese mape "path2" i
naziva datoteke odredjenog lezaja
            'naziv datoteke sastoji se od "doc2" teksta (definicija prvog
lezaja), promjera lezaja i ekstenzije ".txt"
            docfull = path2 & doc1 & (CInt(pro(bl1, dr))) & ".txt"
            'otvara datoteku adresiranu adresom docfull i ucitava velicine
koje se nalaze unutar same
            FileOpen(1, docfull, OpenMode.Input)
            i = 0
            Do Until EOF(1)
                Input(1, sila_l(i))
                Input(1, stsila_l(i))
                Input(1, f_l(i))
                i = i + 1
            Loop
            FileClose(1) 'zatvara datoteku
            docfull = path2 & doc2 & CStr(CInt(pro(bl2, dr))) & ".txt"
'zadavanje adrese za  drugi lezaj
            FileOpen(1, docfull, OpenMode.Input)
            i = 0
            Do Until EOF(1)
                Input(1, sila_d(i))
                Input(1, stsila_d(i))
                Input(1, f_d(i))
                i = i + 1
            Loop
            FileClose(1)
        Catch ex As Exception 'ako dodje do greske u unosu javlja se
navedena poruka te se prekida program
            MessageBox.Show("Proracun nije ispravan. Nema adekvatnog ležaja
u bazi podataka. Molim nadopunite bazu ili promijenite dimenzije vratila")
            Me.Close()
        End Try
    End Sub
    Private Sub odabirbrojaradnihsati()
        'Funkcija za izbor ocekivanih radnih sati ležaja
        Try
            nm = CDbl(TbBrzinaVrtnje.Text)
        Catch ex As Exception
            MessageBox.Show("U polje <Brzina vrtnje> nije uneseni decimalni
broj")
        End Try
    End Sub
    Private Sub PodjelaVratilaNaPolja()
        'Funkcija za podjelu vratila na segmente i dodavanje promjera za
određeno polje te izracunavanje momenta tromosti i mase odredjenog polja
        Dim i As Integer
        p1 = 0
        For i = 0 To nr(dr) - 1 'for petlja se krece od prvog do
posljednjeg segmenta i izvrsava naredbe unutar nje
            If i = bl1 Then 'ako je segment definiran kao segment na kojem
se nalazi prvi lezaj onda
```

```
                bll = d_s(i, dr) - f_l(j)
                s_s = bll / 10
                p1 = p1 + 11 'stvara se 11 novih polja vratila u brojacu
(1+10; jedno polje na kojemu je lezaj te ostalih 10)
                l1(dr) = p1 - 1
                For z = p1 - 11 To p1 - 1
                    p_s(z, dr) = s_s 'definiranje sirine ostalih 10 polja
                    pr_seg(z, dr) = pro(i, dr) 'definiranje promjera polja
                    mi(z, dr) = Math.Pow(pr_seg(z, dr), 3) * Math.PI / 32
'definiranje momenta otpora polja
                    masa(z, dr) = Math.Pow(pr_seg(z, dr) / 2, 2) * Math.PI
* p_s(z, dr) * 7850 / Math.Pow(10, 9) 'definiranje mase polja
                    If i >= UnosTorMomenta.a - 1 And i <= UnosTorMomenta.b
- 1 Then 'ako je segment unutar granica koje su definirane u formi
UnosTorMomenta onda:
                        bool(z, dr) = True
                    End If
                    seg(z, dr) = i 'polje se nalazi na segmentu
                Next z
                p_s(p1 - 1, dr) = f_l(j) 'definiranje sirine polja na
kojemu je lezaj
            ElseIf i = bl2 Then 'ako je segment definiran kao segment na
kojem se nalazi drugi lezaj onda
                bll2 = d_s(i, dr) - f_d(k)
                ss = bll2 / 10
                p1 = p1 + 11
                l2(dr) = p1 - 11
                For z = p1 - 11 To p1 - 1
                    p_s(z, dr) = ss
                    pr_seg(z, dr) = pro(i, dr)
                    mi(z, dr) = Math.Pow(pr_seg(z, dr), 3) * Math.PI / 32
                    masa(z, dr) = Math.Pow(pr_seg(z, dr) / 2, 2) * Math.PI
* p_s(z, dr) * 7850 / Math.Pow(10, 9)
                    If i >= UnosTorMomenta.a - 1 And i <= UnosTorMomenta.b
- 1 Then
                        bool(z, dr) = True
                    End If
                    seg(z, dr) = i
                Next z
                p_s(p1 - 11, dr) = f_d(k)
            ElseIf d_s(i, dr) <= 20 And i <> bl1 And i <> bl2 Then 'ako se
na polju ne nalazi lezaj, a duljina mu je manja od 20 mm
                p1 = p1 + 3 'dodaj 3 nova polja u brojac
                For z = p1 - 3 To p1 - 1
                    p_s(z, dr) = d_s(i, dr) / 3
                    pr_seg(z, dr) = pro(i, dr)
                    mi(z, dr) = Math.Pow(pr_seg(z, dr), 3) * Math.PI / 32
                    masa(z, dr) = Math.Pow(pr_seg(z, dr) / 2, 2) * Math.PI
* p_s(z, dr) * 7850 / Math.Pow(10, 9)
                    If i >= UnosTorMomenta.a - 1 And i <= UnosTorMomenta.b
- 1 Then
                        bool(z, dr) = True
                    End If
                    seg(z, dr) = i
                Next z
            ElseIf d_s(i, dr) > 20 And d_s(i, dr) <= 40 And i <> bl1 And i
<> bl2 Then 'ako se na polju ne nalazi lezaj, a duljina mu je (20 mm < L <
40 mm)
                p1 = p1 + 5 'dodaj 5 novih polja u brojac
                For z = p1 - 5 To p1 - 1
                    p_s(z, dr) = d_s(i, dr) / 5
```

```
                    pr_seg(z, dr) = pro(i, dr)
                    mi(z, dr) = Math.Pow(pr_seg(z, dr), 3) * Math.PI / 32
                    masa(z, dr) = Math.Pow(pr_seg(z, dr) / 2, 2) * Math.PI
* p_s(z, dr) * 7850 / Math.Pow(10, 9)
                    If i >= UnosTorMomenta.a - 1 And i <= UnosTorMomenta.b
- 1 Then
                        bool(z, dr) = True
                    End If
                    seg(z, dr) = i
                Next z
            ElseIf d_s(i, dr) > 40 And d_s(i, dr) <= 100 And i <> bl1 And i
<> bl2 Then 'ako se na polju ne nalazi lezaj, a duljina mu je (40 mm < L <
100 mm)
                p1 = p1 + 8 'dodaj 8 novih polja u brojac
                For z = p1 - 8 To p1 - 1
                    p_s(z, dr) = d_s(i, dr) / 8
                    pr_seg(z, dr) = pro(i, dr)
                    mi(z, dr) = Math.Pow(pr_seg(z, dr), 3) * Math.PI / 32
                    masa(z, dr) = Math.Pow(pr_seg(z, dr) / 2, 2) * Math.PI
* p_s(z, dr) * 7850 / Math.Pow(10, 9)
                    If i >= UnosTorMomenta.a - 1 And i <= UnosTorMomenta.b
- 1 Then
                        bool(z, dr) = True
                    End If
                    seg(z, dr) = i
                Next z
            ElseIf d_s(i, dr) > 100 And i <> bl1 And i <> bl2 Then 'ako se
na polju ne nalazi lezaj, a duljina mu je (100 mm < L)
                p1 = p1 + 12 'dodaj 12 novih polja u brojac
                For z = p1 - 12 To p1 - 1
                    p_s(z, dr) = d_s(i, dr) / 12
                    pr_seg(z, dr) = pro(i, dr)
                    mi(z, dr) = Math.Pow(pr_seg(z, dr), 3) * Math.PI / 32
                    masa(z, dr) = Math.Pow(pr_seg(z, dr) / 2, 2) * Math.PI
* p_s(z, dr) * 7850 / Math.Pow(10, 9)
                    If i >= UnosTorMomenta.a - 1 And i <= UnosTorMomenta.b
- 1 Then
                        bool(z, dr) = True
                    End If
                    seg(z, dr) = i
                Next z
            End If
            For z = 0 To UnosDodatnihSila.i - 1 'Od 0 do "broj segmenata
opterecenih dodatnom radijalnom silom" - 1. Petlja se ponovi onoliko puta
koliko je definirano dodatnih radijalnih sila.
                'ako je, u formi "UnosDodatnihSila", definirani indeks
segmenta jednak indeksu aktualnog segmenta u petlji, onda je iznos dodatne
sile na posljednjem polju segmenta jednak velicini unesenoj u prije
spomenutoj formi
                If UnosDodatnihSila.a(z) - 1 = i Then fdod(p1 - 1, dr) =
UnosDodatnihSila.b(z)
            Next
        Next i
        p11(dr) = p1 'ukupni broj polja
    End Sub
    'Funkcija koja je preuzeta iz gotovog koda
    'Njeno objašnjenje nalazi se u dobivenoj dokumentaciji.
    'Ona proracunava progib, nagib, poprecnu silu i moment savijanja na
svakom polju vratila.
    'dodatno je uneseni kod za proracun maksimalnih naprezanja
    Private Sub PreuzetiDio()
```

```
        'U ovih par redova koda izjednacujemo nove varijable sa varijablama
preuzetih iz koda za proracun prijenosnim matricama
        EY = 210000
        Dim i As Integer
        mm = p11(dr)
        jx = 2
        NP(1) = l1(dr)
        NP(2) = l2(dr)
        CRU(1) = 60656.160156 'definiranje krutosti prvog lezaja
        CRU(2) = 60696.242188 'definiranje krutosti drugog lezaja

        For i = 1 To mm
            EL(i) = p_s(i - 1, dr)
            TJ(i) = mi(i - 1, dr)
            SILA(i) = masa(i - 1, dr) * 9.81 + fdod(i - 1, dr)
        Next i
        For JT = 1 To jx
            i = NP(JT)
            C(i) = CRU(JT)
        Next JT
        V(1) = SILA(1)
        U(1) = 0
        T(1) = -C(1)
        S(1) = V(1) * EL(1) / 2
        R(1) = 0
        AK(1) = T(1) * EL(1)
        Y(1) = S(1) * EL(1) / (3 * EY * TJ(1))
        H(1) = 1
        F(1) = AK(1) * EL(1) / (2 * EY * TJ(1))
        D(1) = Y(1) * EL(1) / 4
        B(1) = EL(1)
        A(1) = 1 + F(1) * EL(1) / 3
        'MN = 1 + 1
        For i = 2 To mm
            'DO 70 I=MN,MM
            S1(i) = SILA(i) * EL(i) / 2
            AK1(i) = -C(i) * EL(i)
            Y1(i) = S1(i) * EL(i) / (3 * EY * TJ(i))
            F3(i) = EL(i) / (EY * TJ(i))
            F1(i) = AK1(i) * EL(i) / (2 * EY * TJ(i))
            D1(i) = Y1(i) * EL(i) / 4
            A3(i) = F3(i) * EL(i) / 2
            A4(i) = A3(i) * EL(i) / 3
            A1(i) = 1 + F1(i) * EL(i) / 3
            A(i) = A(i - 1) * A1(i) + F(i - 1) * EL(i) + AK(i - 1) * A3(i)
+ T(i - 1) * A4(i)
            B(i) = B(i - 1) * A1(i) + H(i - 1) * EL(i) + R(i - 1) * A3(i) +
U(i - 1) * A4(i)
            D(i) = D(i - 1) * A1(i) + Y(i - 1) * EL(i) + S(i - 1) * A3(i) +
V(i - 1) * A4(i) + D1(i)
            F(i) = A(i - 1) * F1(i) + F(i - 1) + AK(i - 1) * F3(i) + T(i -
1) * A3(i)
            H(i) = B(i - 1) * F1(i) + H(i - 1) + R(i - 1) * F3(i) + U(i -
1) * A3(i)
            Y(i) = D(i - 1) * F1(i) + Y(i - 1) + S(i - 1) * F3(i) + V(i -
1) * A3(i) + Y1(i)
            AK(i) = A(i - 1) * AK1(i) + AK(i - 1) + T(i - 1) * EL(i)
            R(i) = B(i - 1) * AK1(i) + R(i - 1) + U(i - 1) * EL(i)
            S(i) = D(i - 1) * AK1(i) + S(i - 1) + V(i - 1) * EL(i) + S1(i)
            T(i) = -A(i - 1) * C(i) + T(i - 1)
            U(i) = -B(i - 1) * C(i) + U(i - 1)
```

```
            V(i) = -D(i - 1) * C(i) + V(i - 1) + SILA(i)
        Next i
        AKO = T(mm) * R(mm) - U(mm) * AK(mm)
        ALI = U(mm)
        WO = 0
        PO = 0
        If AKO <> 0 Then
            WO = (S(mm) * U(mm) - V(mm) * R(mm)) / AKO
        End If
        If ALI <> 0 Then
            PO = -(V(mm) + T(mm) * WO) / ALI
        End If
        'IF(AKO.NE.0.) WO=(S(MM)*U(MM)-V(MM)*R(MM))/AKO
        'IF(ALI.NE.0.) PO=-(V(MM)+T(MM)*WO)/ALI
        For i = 1 To mm
            'DO 80 I=M1,MM
            W(i, dr) = WO * A(i) + PO * B(i) + D(i)  'definicija progiba
*(za odredjeno polje u vratilu)
            P(i, dr) = WO * F(i) + PO * H(i) + Y(i) 'definicija nagiba *
            P(i, dr) = P(i, dr) * 3437.75
            OM(i, dr) = WO * AK(i) + PO * R(i) + S(i) 'definicija momenta
savijanja *
            Q(i, dr) = (WO * T(i) + PO * U(i) + V(i)) 'definicija poprecne
sile *
            smax(i, dr) = OM(i, dr) / mi(i - 1, dr) * Lk(i - 1, dr)
'definicija maksimalnog naprezanja na savijanje *
            tmax(i, dr) = Q(i, dr) / (pr_seg(i - 1, dr) ^ 2) * Math.PI / 4
'definicija maksimalnog naprezanja na smik *
            ttmax(i, dr) = UnosTorMomenta.TM / (2 * mi(i - 1, dr)) * Lk2(i
- 1, dr) 'definicija maksimalnog naprezanja na uvijanje *
        Next i
    End Sub
    Private Sub BtnProracunaj_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles BtnProracun.Click
        'Osnovi proracun kojega pokrece Button "BtnProracunaj"
        Try
            Dim j1, k1 As Integer 'j1 i k1 su pomocni brojaci za DO WHILE
petlju. Oni nam omogucavaju da se naredbe u IF petljama unutar DO WHILE
petlje na j i k brojac odreze tek nakon LOOP-a, sto nam je potrebno za
definiranje uvjeta u LOOP retku.
            ObradaIzboraMaterijala() 'redak u kojemu se poziva navedena
funkcija
            UnosDodatnihSila.ShowDialog()
            UnosTorMomenta.ShowDialog()
            odabirbrojaradnihsati()
            PomocZaOdabirLezaja()
            For dr = 0 To Datagridview.DgwLista.ColumnCount - 1 'petlja
koja provodi kod unutar sebe za svako vratilo u seriji
                j = 0 'resetiranje brojaca za novo vratilo
                k = 0
                j1 = 0
                k1 = 0
                odabirlezaja() 'odabir lezaja za novo vratilo
                'Do While petlja kontrolira da li proraèunata dinamièka
nosivost ležaja zadovoljava dozvoljenu dinamièku nosivost izabranog ležaja.
                'U slucaju da širina ležaja ne zadovoljava petlja bira širi
ležaj.
                Try 'probaj provesti naredbe unutar TRY funkcije
                    Do 'radi i ponavljaj sljedece naredbe, sve dok se ne
ispune uvjeti u LOOP retku
                        j = j1
```

```
                        k = k1
                        PodjelaVratilaNaPolja()
                        ZarezniFaktor1()
                        ZarezniFaktor2()
                        PreuzetiDio()
                        ProracunLezajaL()
                        ProracunLezajaD()
                        If sila_l(j) <= Math.Abs(Pl(dr)) Then
                            j1 = j + 1 'ako dopusteno naprezanje ne
zadovoljava povecaj brojac
                        End If
                        If sila_d(k) <= Math.Abs(Pd(dr)) Then
                            k1 = k + 1
                        End If
                    Loop Until sila_l(j) >= Math.Abs(Pl(dr)) And sila_d(k)
>= Math.Abs(Pd(dr)) 'ponavljaj sve dok dopusteno naprezanje nije vece od
dinamickog opterecenja
                Catch ex As Exception 'ako je javi greska unutar TRY
funkcije napravi sljedece
                    If j1 > sila_l.Length Then
                        MessageBox.Show("Ne postoji " &
CbTipLezajaPrvi.Text & " sa zadovoljavajucom dinamickom cvrstocom za
vratilo broj " & dr)
                    Else
                        MessageBox.Show("Ne postoji " &
CbTipLezajaDrugi.Text & " sa zadovoljavajucom dinamickom cvrstocom za
vratilo broj " & dr)
                    End If
                    Me.Close() 'prekid programa
                End Try
                DefinicijaC0iC()
            Next
            txt_Fa.Enabled = False
            TxtBrojradnihsati.Enabled = False
            TbBrSegDrugog.Enabled = False
            TbBrSegPrvog.Enabled = False
            TbBrzinaVrtnje.Enabled = False
            CbIzborMaterijala.Enabled = False
            CbTipLezajaDrugi.Enabled = False
            CbTipLezajaPrvi.Enabled = False
            BtnIspisUExcel.Enabled = True
            CheckBox1.Enabled = False
            BtnProracun.Enabled = False
            MessageBox.Show("Proracun je zavrsen. Pritiskom na >Ispis u
Excel< dobivate podatke o vratilu u biranoj .xls datoteci.", "Kraj")
        Catch ex As Exception
            MsgBox("Došlo je do greške u unosu. Molim ponovite čitavi
postupak. Ako se greška ponovi, zatražite pomoć stručne osobe.")
            Me.Close()
        End Try
    End Sub
    'sljedeca dva retka su dio koda za gasenje svih zaostalih EXCEL.EXE
datoteka u "Processes" direktoriju
    Declare Function PostMessage Lib "user32" Alias "PostMessageA" (ByVal
hwnd As Int32, ByVal wMsg As Int32, ByVal wParam As Int32, ByVal lParam As
Int32) As Int32
    Const WM_QUIT = &H12
    Dim oExcel As Microsoft.Office.Interop.Excel.Application = New
Microsoft.Office.Interop.Excel.Application
    Private Sub BtnIspisUExcel_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles BtnIspisUExcel.Click
```

```
        'BtnIspisUExcel Button služi za ubacivanje podataka u Excel
datoteku.
        Dim i, j, n As Integer 'i,j = lokalni brojaci; n = broj sheet-ova u
odabranom dokumentu
        Dim a As Double 'koordinata vratila na osi apscisa
        Dim oWorkBook As Microsoft.Office.Interop.Excel.Workbook 'varijabla
koja definira Excel WorkBook
        Dim oWorkSheet As Microsoft.Office.Interop.Excel.Worksheet
'varijabla koja definira Excel WorkSheet
        Dim fileD As New OpenFileDialog()
        fileD.ShowDialog() 'otvara prozor za odabir datoteke
        Dim fileName As String = fileD.FileName
        oWorkBook = oExcel.Workbooks.Open(fileName) 'oWorkBook = odabrani
Excel WorkBook
        VelicineZaDijagram() 'poziv na funkciju
        For j = 0 To Datagridview.DgwLista.ColumnCount - 1 'unos podataka u
excel datoteku
            a = 0
            n = oWorkBook.Sheets.Count 'definiranje broja sheet-ova
            oWorkSheet = oWorkBook.Worksheets().Item(2) '"aktiviranje"
drugog sheet-a kao objekta za koristenje
            oWorkSheet.Activate()
            oWorkSheet.Copy(, oWorkBook.Worksheets().Item(n)) 'kopiranje
referentnog radnog lista na zadnje mijesto
            n = n + 1
            oWorkSheet = oWorkBook.Worksheets().Item(n) 'novonastali sheet
postaje "aktivan"
            oWorkSheet.Name = "Vratilo " & n - 2 'promjena imena
            oWorkSheet.Cells(1, 1).Value = "Vratilo " & n - 2 'promjena
naslova
            oWorkSheet.Cells(5, 5).Value = bl1 + 1 'dodavanje vrijednosti u
polja
            oWorkSheet.Cells(5, 6).Value = bl2 + 1
            oWorkSheet.Cells(6, 5).Value = pr_seg(l1(j), j)
            oWorkSheet.Cells(6, 6).Value = pr_seg(l2(j), j)
            oWorkSheet.Cells(7, 5).Value = p_s(l1(j), j)
            oWorkSheet.Cells(7, 6).Value = p_s(l2(j), j)
            oWorkSheet.Cells(8, 5).Value = IspisC0l(j)
            oWorkSheet.Cells(8, 6).Value = IspisC0d(j)
            oWorkSheet.Cells(9, 5).Value = IspisCl(j)
            oWorkSheet.Cells(9, 6).Value = IspisCd(j)
            oWorkSheet.Cells(10, 5).Value = Math.Abs(Pl(j))
            oWorkSheet.Cells(10, 6).Value = Math.Abs(Pd(j))
            oWorkSheet.Cells(11, 5).Value = Math.Abs(Lhl(j))
            oWorkSheet.Cells(11, 6).Value = Math.Abs(Lhd(j))
            For i = 1 To p11(j)
                a = a + p_s(i - 1, j)
                oWorkSheet.Cells(i + 15, 1).Value = a
                oWorkSheet.Cells(i + 15, 2).Value = p_s(i - 1, j)
                oWorkSheet.Cells(i + 15, 3).Value = pr_seg(i - 1, j)
                oWorkSheet.Cells(i + 15, 4).Value = OM(i, j)
                oWorkSheet.Cells(i + 15, 6).Value = smax(i, j)
                oWorkSheet.Cells(i + 15, 8).Value = Q(i, j)
                oWorkSheet.Cells(i + 15, 10).Value = tmax(i, j)
                If bool(i - 1, j) = True Then
                    oWorkSheet.Cells(i + 15, 12).Value = UnosTorMomenta.TM
                    oWorkSheet.Cells(i + 15, 14).Value = ttmax(i, j)
                Else
                    oWorkSheet.Cells(i + 15, 12).Value = 0
                    oWorkSheet.Cells(i + 15, 14).Value = 0
                End If
```

```
                oWorkSheet.Cells(i + 15, 16).Value = W(i, j)
                oWorkSheet.Cells(i + 15, 18).Value = P(i, j)
            Next
            For i = 0 To p11(j) - 1
                oWorkSheet.Cells(ca(i, j) + 15, 5).Value = OM(i + 1, j)
                oWorkSheet.Cells(ca(i, j) + 15, 7).Value = smax(i + 1, j)
                oWorkSheet.Cells(ca(i, j) + 15, 9).Value = Q(i + 1, j)
                oWorkSheet.Cells(ca(i, j) + 15, 11).Value = tmax(i + 1, j)
                If bool(i, j) = True Then
                    oWorkSheet.Cells(ca(i, j) + 15, 13).Value =
UnosTorMomenta.TM
                    oWorkSheet.Cells(ca(i, j) + 15, 15).Value = ttmax(i +
1, j)
                Else
                    oWorkSheet.Cells(ca(i, j) + 15, 13).Value = 0
                    oWorkSheet.Cells(ca(i, j) + 15, 15).Value = 0
                End If
                oWorkSheet.Cells(ca(i, j) + 15, 17).Value = W(i + 1, j)
                oWorkSheet.Cells(ca(i, j) + 15, 19).Value = P(i + 1, j)
            Next
            For i = 0 To di(j) - 1
                oWorkSheet.Cells(i + 16, 20).Value = XOs(i, j)
            Next
        Next
        Try
            oWorkBook.Application.DisplayAlerts = False
            oWorkBook.SaveAs(fileName) 'sprema dokument
            oWorkBook.Close()
            TryToDisplayGeneratedFile(fileName) 'poziv na funkciju
        Catch
            MsgBox("Zatvorite sve aktivne .xls datoteke.")
        End Try
    End Sub
    Sub TryToDisplayGeneratedFile(ByVal fileName As String)
        'funkcija koja otvara Microsoft Excel te gore uredjenu datoteku
        Try
            System.Diagnostics.Process.Start(fileName)
        Catch
            Console.WriteLine(fileName + " created in application folder.")
        End Try
    End Sub
    Private Sub ProracunVratila_FormClosed(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.FormClosedEventArgs) Handles
MyBase.FormClosed
        'ostatak koda koji sluzi za gasenje svih zaostalih EXCEL.EXE
datoteka u "Processes" direktoriju
        oExcel.Quit()
        PostMessage(oExcel.Hwnd, WM_QUIT, 0, 0)
    End Sub
    Private Sub PomocZaOdabirLezaja()
        'PomocZaOdabirLezaja() stvara adresu mape u kojoj se nalazi
datoteke s lezajima
        Dim i As Integer
        Dim Open As New OpenFileDialog 'definira "Open" formu
        MessageBox.Show("Odaberite bilo koji ležaj iz baze. Odabir služi za
referiranje na pravilnu adresu. Nije nuzno da ce se isti lezaj koristiti u
proracunu.")
        Open.ShowDialog() 'aktiviranje "Open" forme
        Dim path As String = Open.FileName 'varijabla "path" poprima
vrijednost adrese odabrane datoteke (lezaja)
        Dim slash As New Char
```

```
        slash = "\"
        For i = 0 To path.Length - 1
            If path.Chars(i) = slash Then
                n = i 'definira se indeks tj. pozicija zadnje "\"
(backslash)
            End If
        Next
        path2 = path.Remove(n + 1, path.Length - (n + 1)) 'Brise adresu do
gore definirane pozicije tj. brise ime lezaja. Time dobivamo adresu mape.
    End Sub
    Private Sub ProracunLezajaD()
        'proracun vijeka trajanja drugog lezaja te iznosa dinamickog
opterecenja na lezaj
        Dim i, j, z, C0 As Integer
        Dim C, Fr, Fa, X1, Y1, omr, er As Double
        'i,j,z - brojaci
        'X1 - radijalni faktor lezaja
        'Y1 - aksialni faktor lezaja
        'omr - varijabla koja oznacava omjer (Fa / C0)
        Dim om(10), el(10), Y(10) As Double
        Dim test As Boolean
        test = False
        C0 = stsila_d(k) 'dozvoljeno staticko naprezanje
        C = sila_d(k) 'dozvoljeno dinamicko naprezanje
        Fr = Q(l2(dr), dr) 'radijalna sila na lezaj
        Fa = txt_Fa.Text 'aksialna sila na lezaj
        'unos potrebnih koeficijenata iz definiranih tablica
        For i = 0 To ListBoxOM.Items.Count - 1
            om(i) = ListBoxOM.Items.Item(i)
        Next
        For i = 0 To ListBoxEL.Items.Count - 1
            el(i) = ListBoxEL.Items.Item(i)
        Next
        For i = 0 To ListBoxY.Items.Count - 1
            Y(i) = ListBoxY.Items.Item(i)
        Next
        omr = Fa / C0 'varijabla koja oznacava ovdje prikazani omjer
        If omr >= 0.025 And omr <= 0.5 Then 'vrijednost Y u jednadzbi
Pd(dr) = X1 * Fr + Y1 * Fa moze se izracunati samo ako je "omr" u
definiranim granicama
            For i = 0 To ListBoxOM.Items.Count - 1
                If omr <> om(i) And omr > om(i) Then
                    j = j + 1
                    er = (el(j) - el(j - 1)) / (om(j) - om(j - 1)) * (omr -
om(j - 1)) + el(j - 1) 'aproksimacija velicine "er" s obzirom na tablicu
                ElseIf omr = om(i) Then
                    er = el(i)
                    z = i
                    test = True 'ako er nije potrebno aproksimiati test
varijabla je "TRUE"
                End If
            Next
            If Fa / Fr <= er Then
                X1 = 1
                Y1 = 0
            ElseIf Fa / Fr > er Then
                X1 = 0.56
                If test = True Then 'ako se nije aproksimirala varijabla
"er", ne treba aproksimirati ni "Y1"
                    Y1 = Y(z)
                Else
```

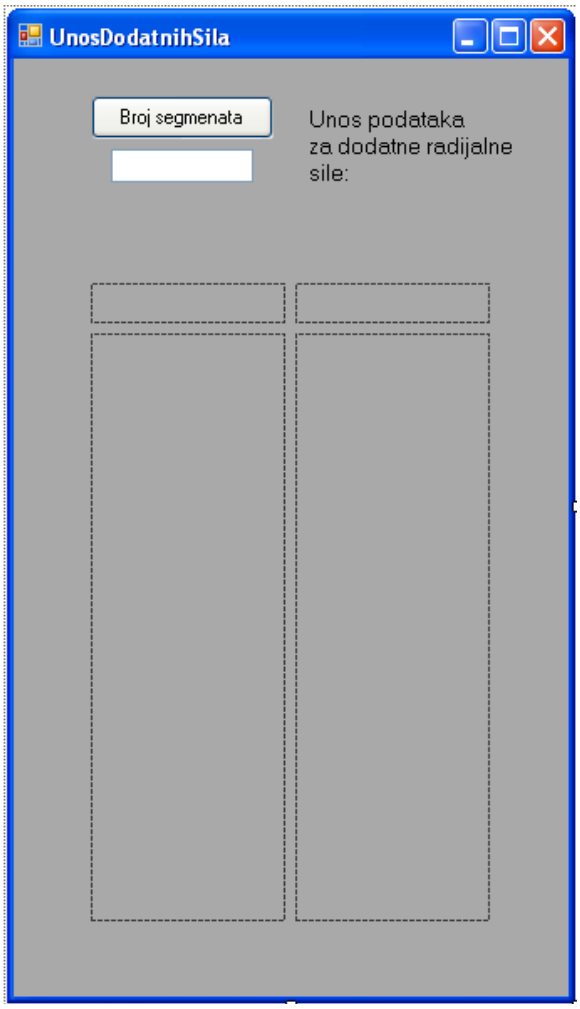
```
                    Y1 = (Y(j) - Y(j - 1)) / (om(j) - om(j - 1)) * (omr -
om(j - 1)) + Y(j - 1) 'aproksimacija velicine "Y1"
                End If
            End If
            Pd(dr) = X1 * Fr + Y1 * Fa 'izracun dinamickog opterecenja
        Else
            Pd(dr) = Fr
        End If
        Lhd(dr) = 1000000 / (60 * nm) * ((C / Pd(dr)) ^ 3) 'izracun vijeka
trajanja lezaja
        i = i + 1
    End Sub
    Private Sub ProracunLezajaL()
        'proracun vijeka trajanja prvog lezaja te iznosa dinamickog
opterecenja na lezaj
        Dim i, j1, z, C0 As Integer
        Dim C, Fr, Fa, X1, Y1, omr, er As Double
        Dim om(10), el(10), Y(10) As Double
        Dim test As Boolean
        test = False
        j1 = 0
        C0 = stsila_l(j)
        C = sila_l(j)
        Fr = Q(l1(dr), dr)
        Fa = txt_Fa.Text
        For i = 0 To ListBoxOM.Items.Count - 1
            om(i) = ListBoxOM.Items.Item(i)
        Next
        For i = 0 To ListBoxEL.Items.Count - 1
            el(i) = ListBoxEL.Items.Item(i)
        Next
        For i = 0 To ListBoxY.Items.Count - 1
            Y(i) = ListBoxY.Items.Item(i)
        Next
        omr = Fa / C0
        If omr >= 0.025 And omr <= 0.5 Then
            For i = 0 To ListBoxOM.Items.Count - 1
                If omr <> om(i) And omr > om(i) Then
                    j1 = j1 + 1
                    er = (el(j1) - el(j1 - 1)) / (om(j1) - om(j1 - 1)) *
(omr - om(j1 - 1)) + el(j1 - 1)
                ElseIf omr = om(i) Then
                    er = el(i)
                    z = i
                    test = True
                End If
            Next
            If Fa / Fr <= er Then
                X1 = 1
                Y1 = 0
            ElseIf Fa / Fr > er Then
                X1 = 0.56
                If test = True Then
                    Y1 = Y(z)
                Else
                    Y1 = (Y(j1) - Y(j1 - 1)) / (om(j1) - om(j1 - 1)) * (omr
- om(j1 - 1)) + Y(j1 - 1)
                End If
            End If
            Pl(dr) = X1 * Fr + Y1 * Fa 'izracun dinamickog opterecenja
        Else
```

```
            Pl(dr) = Fr
        End If
        Lhl(dr) = 1000000 / (60 * nm) * ((C / Pl(dr)) ^ 3) 'izracun vijeka
trajanja lezaja
        i = i + 1
    End Sub
    Private Sub ZarezniFaktor1()
        'funkcija koja proracunava faktore zareznog djelovanja kod
savijanja na poljima koja sadrze prijelazne radijuse
        Dim i As Integer
        Dim t, r, a As Double
        't - razlika polumjera susjednih segmenata
        'r - radijus zaobljenja
        'a - radijos manjeg (ovisno o promjeru) segmenta
        For i = 0 To p11(dr) - 1 'za sva polja koja nemaju prijelazni
radijus faktor je !
            Lk(i, dr) = 1
        Next
        For i = 1 To p11(dr) - 1 'izracun faktora prema definiranim
formulama
            If i <> 0 Then
                If pr_seg(i, dr) > pr_seg(i - 1, dr) Then
                    a = pr_seg(i - 1, dr) / 2
                    t = (pr_seg(i, dr) - pr_seg(i - 1, dr)) / 2
                    r = rad(seg(i - 1, dr), dr)
                    Lk(i - 1, dr) = 1 + (1 / Math.Sqrt((0.77 / (t / r)) +
3.8 * ((1 + a / r) ^ 2) / (a / r) ^ 3) + (0.2 / (t / r) ^ 3) * (a / r) / (a
/ r + t / r))
                ElseIf pr_seg(i, dr) < pr_seg(i - 1, dr) Then
                    a = pr_seg(i, dr) / 2
                    t = (pr_seg(i - 1, dr) - pr_seg(i, dr)) / 2
                    r = rad(seg(i, dr), dr)
                    Lk(i, dr) = 1 + (1 / Math.Sqrt((0.77 / (t / r)) + 3.8 *
((1 + a / r) ^ 2) / (a / r) ^ 3) + (0.2 / (t / r) ^ 3) * (a / r) / (a / r +
t / r))
                ElseIf pr_seg(i, dr) = pr_seg(i - 1, dr) And p_s(i, dr) <>
p_s(i - 1, dr) And i - 1 <> l1(dr) And i - 1 <> l2(dr) Then
                    a = pr_seg(i, dr) / 2
                    t = rad(seg(i, dr), dr)
                    r = rad(seg(i, dr), dr)
                    Lk(seg(i, dr), dr) = 1 + (1 / Math.Sqrt((0.25 / (t /
r)) + 1.8 * ((1 + a / r) ^ 2) / (a / r) ^ 3))
                    Lk(seg(i - 1, dr), dr) = 1 + (1 / Math.Sqrt((0.25 / (t
/ r)) + 1.8 * ((1 + a / r) ^ 2) / (a / r) ^ 3))
                End If
            End If
        Next
    End Sub
    Private Sub ZarezniFaktor2()
        'funkcija koja proracunava faktore zareznog djelovanja kod torzije
na poljima koja sadrze prijelazne radijuse
        Dim i As Integer
        Dim t, r, a As Double
        't - razlika polumjera susjednih segmenata
        'r - radijus zaobljenja
        'a - radijos manjeg (ovisno o promjeru) segmenta
        For i = 0 To p11(dr) - 1
            If bool(i, dr) = True Then
                Lk2(i, dr) = 1
                If i <> 0 Then
```

```
                    If pr_seg(i, dr) > pr_seg(i - 1, dr) Then 'izracun
faktora prema definiranim formulama
                        a = pr_seg(i - 1, dr) / 2
                        t = (pr_seg(i, dr) - pr_seg(i - 1, dr)) / 2
                        r = rad(seg(i - 1, dr), dr)
                        Lk2(i - 1, dr) = 1 + (1 / Math.Sqrt((3.4 / (t / r))
+ 13 * ((1 + a / r) ^ 2) / (a / r) ^ 3) + (1 / (t / r) ^ 2) * (a / r) / (a
/ r + t / r))
                    ElseIf pr_seg(i, dr) < pr_seg(i - 1, dr) Then
                        a = pr_seg(i, dr) / 2
                        t = (pr_seg(i - 1, dr) - pr_seg(i, dr)) / 2
                        r = rad(seg(i, dr), dr)
                        Lk2(i, dr) = 1 + (1 / Math.Sqrt((3.4 / (t / r)) +
13 * ((1 + a / r) ^ 2) / (a / r) ^ 3) + (1 / (t / r) ^ 2) * (a / r) / (a /
r + t / r))
                    ElseIf pr_seg(i, dr) = pr_seg(i - 1, dr) And p_s(i, dr)
<> p_s(i - 1, dr) And i - 1 <> l1(dr) And i - 1 <> l2(dr) Then
                        a = pr_seg(i, dr) / 2
                        t = rad(seg(i, dr), dr)
                        r = rad(seg(i, dr), dr)
                        Lk2(i, dr) = 1 + (1 / Math.Sqrt((1 / (t / r)) + 7 *
((1 + a / r) ^ 2) / (a / r) ^ 3))
                        Lk2(i - 1, dr) = 1 + (1 / Math.Sqrt((1 / (t / r)) +
7 * ((1 + a / r) ^ 2) / (a / r) ^ 3))
                    End If
                End If
            End If
        Next
    End Sub
    Dim IspisCl(50), IspisCd(50), IspisC0l(50), IspisC0d(50) As Integer
    'IspisCl(50) - dozvoljena dinamicka naprezanja na prve lezajeve u
seriji
    'IspisCd(50) - dozvoljena dinamicka naprezanja na druge lezajeve u
seriji
    'IspisC0l(50) - dozvoljena staticka naprezanja na prve lezajeve u
seriji
    'IspisC0d(50) - dozvoljena staticka naprezanja na druge lezajeve u
seriji
    Private Sub DefinicijaC0iC()
        'funkcija sprema dopustena naprezanja svih odabranih lezaja u
analizi
        Dim i, j As Integer
        For j = 0 To Datagridview.DgwLista.ColumnCount - 1
            For i = 0 To f_l.Length - 1
                If f_l(i) = p_s(l1(dr), j) Then
                    IspisCl(j) = sila_l(i)
                    IspisC0l(j) = stsila_l(i)
                End If
            Next
            For i = 0 To f_d.Length - 1
                If f_d(i) = p_s(l2(dr), j) Then
                    IspisCd(j) = sila_d(i)
                    IspisC0d(j) = stsila_d(i)
                End If
            Next
        Next
    End Sub
    Dim XOs(20000, 50) As Double
    Dim ca(1000, 50) As Double
    Private Sub VelicineZaDijagram()
        Dim a As Double
```

```
        For j = 0 To Datagridview.DgwLista.ColumnCount - 1
            For i = 0 To CInt(length(j)) * 2
                XOs(i, j) = i / 2 'definicija polja(XOs =
{0,0.5,1,1.5,..."duljina vratila"})
                di(j) = di(j) + 1
            Next
            a = 0
            For i = 0 To p11(j) - 1
                a = a + p_s(i, j)
                If CInt(a) - a >= 0.26 Then 'definiranje aproksimiranih
koordinata vratila
                    ca(i, j) = (CInt(a) - 0.5) * 2 + 1
                ElseIf CInt(a) - a >= -0.24 And CInt(a) - a < 0.26 Then
                    ca(i, j) = CInt(a) * 2 + 1
                ElseIf CInt(a) - a <= -0.24 Then
                    ca(i, j) = (CInt(a) + 0.5) * 2 + 1
                End If
            Next
        Next
    End Sub
    Private Sub CheckBox1_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CheckBox1.CheckedChanged
        If CheckBox1.Checked = True Then 'ako je "Checkbox1" ukljuceni
Button "Proracunaj" je aktiviran
            BtnProracun.Enabled = True
        Else
            BtnProracun.Enabled = False
        End If
    End Sub
End Class
```

Sučelje za unos dodatnih radijalnih sila



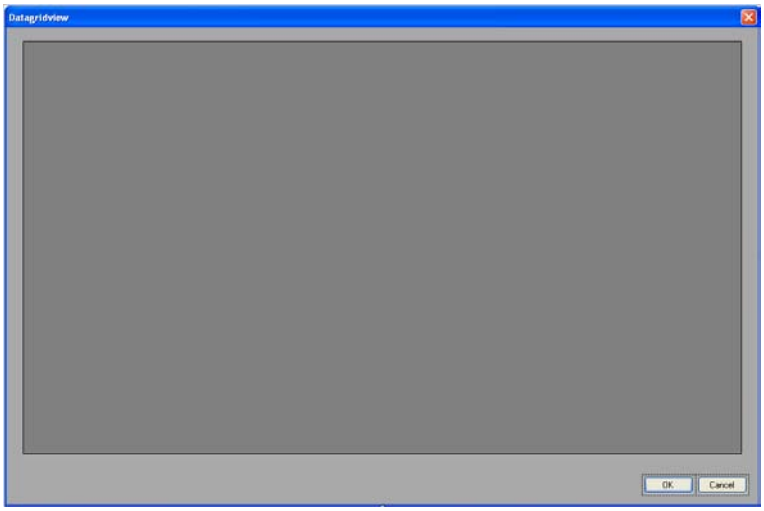
```
Public Class UnosDodatnihSila

    Dim Txtdynamic As TextBox 'Dinamicki kreirani textbox. Omogucava
kreiranje dodatnih textbox-eva na zeljenom mjestu.
    Dim lab1 As Label 'Dinamicki kreirani label. Omogucava kreiranje
dodatnih label-a na zeljenom mjestu.
    Dim j As Integer
    Public i As Integer 'broj segmenata koji su optereceni dodatnom
radijalnom silom
    Public a(100), b(100) As Integer
    'a(100) - polje koje sadrzi indekse segmenata koji su optereceni
dodatnom silom
    'b(100) - polje koje sadrzi iznose dodatnih radijalnih sila na
segmentima
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        If Button1.Text = "Broj segmenata" Then 'prvi dio If funkcije
stvara "i" slobodnih mjesta za unos podataka o silama
            Try
                i = TextBox1.Text
                'naredni dio stvara 2 labela koji opisuju novonastala
slobodna polja
                lab1 = New Label
                lab1.AutoSize = True
                lab1.Font = New System.Drawing.Font("Verdana", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point,
CType(0, Byte))
                lab1.Name = "lab_1"
                lab1.Text = "Broj segmenta: "
                FlowLayoutPanel2.Controls.Add(lab1)
                lab1 = New Label
                lab1.AutoSize = True
                lab1.Font = New System.Drawing.Font("Verdana", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point,
CType(0, Byte))
                lab1.Name = "lab_2"
                lab1.Text = "Iznos sile [N]: "
                FlowLayoutPanel3.Controls.Add(lab1)
                For j = 0 To i - 1  'stvara slobodna polja za unos rednog
broj segmenta
                    Txtdynamic = New TextBox
                    Txtdynamic.AutoSize = True
                    Txtdynamic.Font = New System.Drawing.Font("Verdana",
8.25!, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point,
CType(0, Byte))
                    Txtdynamic.Name = "rednibroj_" & j & "_mycntrl"
                    FlowLayoutPanel1.Controls.Add(Txtdynamic)
                Next
                For j = 0 To i - 1  'stvara slobodna polja za unos iznosa
sila na segmentima
                    Txtdynamic = New TextBox
                    Txtdynamic.AutoSize = True
                    Txtdynamic.Font = New System.Drawing.Font("Verdana",
8.25!, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point,
CType(0, Byte))
                    Txtdynamic.Name = "sila_" & j & "_mycntrl"
                    FlowLayoutPanel4.Controls.Add(Txtdynamic)
                Next
                Button1.Text = "Unesi" 'mjenja tekst Button-a "Button1" u
"Unesi"
```

```
            Catch ex As Exception
                MessageBox.Show("Unesi broj segmenata")
            End Try
        Else
            Try
                For j = 0 To i - 1
                    If FlowLayoutPanel1.Controls(j).Name = "rednibroj_" & j
& "_mycntrl" Then
                        If CInt(Txtdynamic.Text) > 0 Then
                            a(j) = CInt(FlowLayoutPanel1.Controls(j).Text)
'sprema podatke o rednom broju u polje a(100)
                        End If
                    End If
                    If FlowLayoutPanel4.Controls(j).Name = "sila_" & j &
"_mycntrl" Then
                        If CInt(Txtdynamic.Text) > 0 Then
                            b(j) = CInt(FlowLayoutPanel4.Controls(j).Text)
'sprema podatke o iznosu sile u polje b(100)
                        End If
                    End If
                Next
                Me.Hide()
            Catch ex As Exception
                MessageBox.Show("Molimo ispunite sve podatke")  'u slucaju
greske tj. nepotpunog unosa javlja se prikladna poruka
            End Try
        End If
    End Sub
End Class
```

Sučelje za unos torzionog momenta

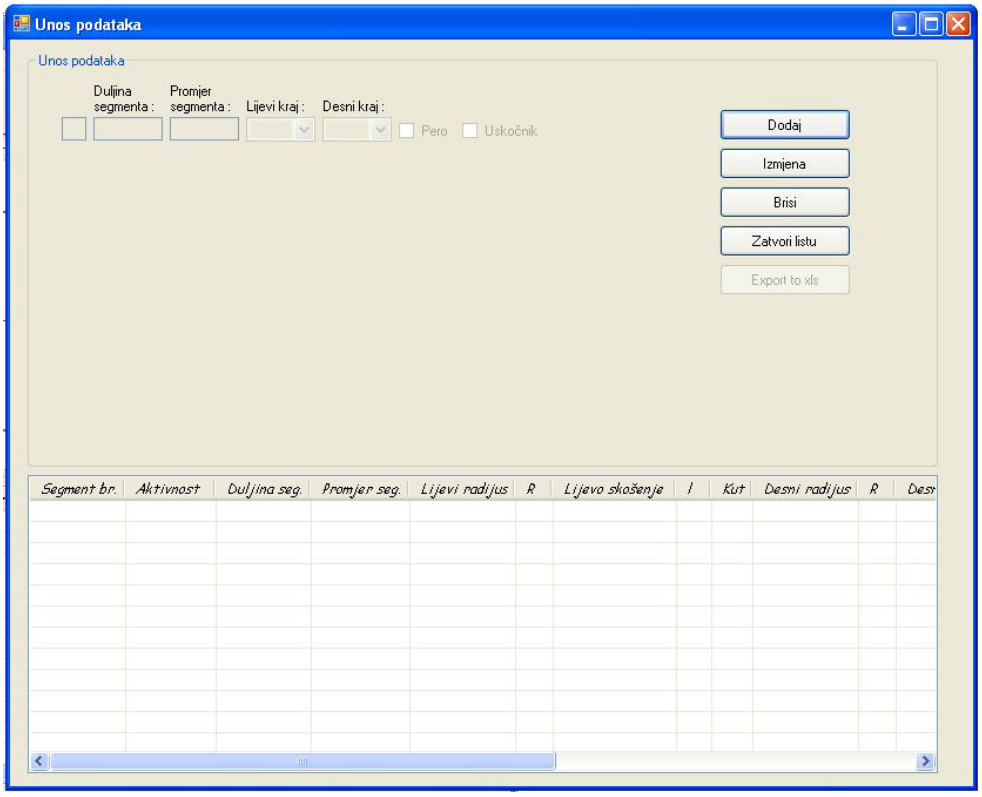
```
Public Class UnosTorMomenta
    Public a, b, TM As Integer 'public varijable su dostupne glavnoj formi
    'VARIJABLE
    'a - prvi segment koji je opterecen momentom
    'b - posljednji segment koji je opterecen momentom
    'TM - iznos torzionog momenta
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Try 'try funkcija osigurava pravilan unos podataka
            a = CInt(TextBox1.Text) 'cijeli broj
            b = CInt(TextBox2.Text) 'cijeli broj
            TM = CDbl(TextBox3.Text) 'decimalni broj
            Me.Hide()
        Catch ex As Exception
            MessageBox.Show("Unos nije ispravan.")
        End Try
    End Sub
      End Class
```

Osnovno sučelje za unos vratila programom u Visual Basic-u



```
Imports System.Windows.Forms
Public Class Datagridview
    Private Sub OK_Button_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles OK_Button.Click
        Me.DialogResult = System.Windows.Forms.DialogResult.OK
        Me.Close()
    End Sub
    Private Sub Cancel_Button_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles Cancel_Button.Click
        Me.DialogResult = System.Windows.Forms.DialogResult.Cancel
        Me.Close()
    End Sub
    Private Sub Datagridview_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Dim i As Integer
        For i = DgwLista.ColumnCount - 1 To 0 Step -1 'petlja brise sve
prazne stupce u Datagridview-u
            If Convert.ToString(DgwLista.Item(i, 0).Value()) = "" Then
                DgwLista.Columns.RemoveAt(i)
            End If
        Next
        For i = DgwLista.RowCount - 1 To 0 Step -1 'petlja brise sve prazne
retke u Datagridview-u
            If Convert.ToString(DgwLista.Item(0, i).Value()) = "" Then
                DgwLista.Rows.RemoveAt(i)
            End If
        Next
    End Sub
End Class
```

Osnovno sučelje za unos vratila programom u Visual Basic-u



```
Public Class Glavna
    ' Definicija varijabli.
    Dim pero_form As New Pero_form 'definira formu Pero_form
    Dim radijus As New Radijus
    Dim skosenje As New Skosenje
    Dim uskocnik_form As New Uskocnik_form
    Dim r1, r2, S1_L, S2_L, S1_D, S2_D As String
    'r1 - lijevi radijus na segmentu
    'r2 - desni radijus na segmentu
    'S1_L - duljina za lijevo skosenje
    'S2_L - kut za lijevo skosenje
    'S1_D - duljina za desno skosenje
    'S2_D - kut za desno skosenje
    Dim b_p, t_p, d, e1 As Double
    'b_p - sirina utora za pero
    't_p - dubina utora za pero
    'd, e1 - dimentije koje definiraju utor za uskocnik
    Dim blnAdd As Boolean
    ' Kod pokretanja cjelokupnog programa onemoguceno je koristenje dole
navedenih elemenata. Oni se aktiviraju kasnijim pritiskom na Button
"Dodaj".
    Private Sub Glavna_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        txt_Lsegmenta.Enabled = False
        txt_Dsegmenta.Enabled = False
        combo_lijevi.Enabled = False
        combo_desni.Enabled = False
        check_pero.Enabled = False
        check_uskocnik.Enabled = False
    End Sub
    ' Naredne dvije funkcije sluze za odgovarajucu izmjenu slika pri
odabiru odredjenog Textbox-a
    Private Sub txt_Lsegmenta_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles txt_Lsegmenta.Click
        PictureBox1.Show()
        PictureBox2.Hide()
    End Sub
    Private Sub txt_Dsegmenta_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles txt_Dsegmenta.Click
        PictureBox1.Hide()
        PictureBox2.Show()
    End Sub
    ' Funkcije "combo_lijevi_SelectionChanged" te
"combo_desni_SelectionChanged" služe za definiciju rubova.
    Private Sub combo_lijevi_SelectionChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
combo_lijevi.SelectedValueChanged
        If combo_lijevi.Text = "Radijus" Then
            radijus.raddesni = False
            radijus.radlijevi = True 'logičke varijable koje su potrebne
jer se u jednoj formi unose podaci za lijevi i desni rub.
            radijus.pbox_lijeva.Show() 'naredba koja manipulira prikaz
slike.
            radijus.ShowDialog() 'otvara formu "Radijus" kao dialog prozor,
sto znaci da treba zatvoriti istu prije nego se moze koristiti bilo sto
drugo.
            r1 = radijus.R 'pridodaje varijabli vrijednost unesenu u formi
"Radijus"
            'daljnje naredbe su istog tipa kao sto i predhodne pa ih ne
treba ponovno objasnjavat
        ElseIf combo_lijevi.Text = "Skošenje" Then
```

```
            skosenje.skodesno = False
            skosenje.skolijevo = True
            skosenje.pbox_lijeva.Show()
            skosenje.ShowDialog()
            S1_L = skosenje.S1
            S2_L = skosenje.S2
        End If
    End Sub
    Private Sub combo_desni_SelectionChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles combo_desni.SelectedValueChanged
        If combo_desni.Text = "Radijus" Then
            radijus.raddesni = True
            radijus.radlijevi = False
            radijus.pbox_desna.Show()
            radijus.ShowDialog()
            r2 = radijus.R
        ElseIf combo_desni.Text = "Skošenje" Then
            skosenje.skodesno = True
            skosenje.skolijevo = False
            skosenje.pbox_desna.Show()
            skosenje.ShowDialog()
            S1_D = skosenje.S1
            S2_D = skosenje.S2
        End If
    End Sub
    Private Sub check_pero_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles check_pero.CheckedChanged
        If check_pero.Checked Then
            pero_form.ShowDialog() 'pokrece formu "Pero_form" u slucaju da
je CheckBox oznacen
        End If
    End Sub
    Private Sub check_uskocnik_CheckedChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
check_uskocnik.CheckedChanged
        If check_uskocnik.Checked Then
            uskocnik_form.ShowDialog()
        End If
    End Sub
    'naredna funkcija se odvija na pritisak Buttona "btn_Dodaj"
    Private Sub btn_dodaj_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_dodaj.Click
        If List_all.Items.Count = 20 Then 'if naredbom smo ogranicili broj
unosa na 20
            MessageBox.Show("Maksimalni broj segmenata je 20")
        Else
            If btn_dodaj.Text.ToLower() = "dodaj" Then 'izmjene koje se
odvijaju u slucaju odabira btn_Dodaj kad na njemu stoji "Dodaj"
                txt_index.Text = List_all.Items.Count + 1
                combo_lijevi.Text = combo_lijevi.Items.Item(0).ToString
                combo_desni.Text = combo_desni.Items.Item(0).ToString
                txt_Lsegmenta.Enabled = True
                txt_Dsegmenta.Enabled = True
                combo_lijevi.Enabled = True
                combo_desni.Enabled = True
                check_pero.Enabled = True
                check_uskocnik.Enabled = True
                check_pero.Checked = False
                check_uskocnik.Checked = False
                btn_dodaj.Text = "Spremi"
                btn_izmjena.Text = "Odustani"
```

```
                btn_brisi.Enabled = False
                txt_Lsegmenta.Clear()
                txt_Dsegmenta.Clear()
                blnAdd = True 'logicka varijabla kojom program prepoznaje
radi li se o unosu ili izmjeni
            Else
                Dim txt1, txt2 As Double 'izmjene koje se odvijaju u
slucaju odabira btn_Dodaj kad na njemu stoji "Spremi"
                Try 'Naredba "Try-Catch ex As Exception- End Try" osigurava
unos decimalnih brojeva u trazena polja.
                    'Stoga se ne moze unijeti tekst.
                    txt_Lsegmenta.BackColor = Color.White
                    txt_Dsegmenta.BackColor = Color.White
                    txt1 = Convert.ToDouble(txt_Lsegmenta.Text) 'ako je
preobrazba u Double uspjesna nastavlja sa spremanjem i ostalim izmjenama
                    ' ako je doslo do greske skace na naredbe unutar "Catch
ex As Exception"
                    txt2 = Convert.ToDouble(txt_Dsegmenta.Text)
                    If blnAdd Then
                        AddItemToListView()
                    Else
                        EditItemInListView()
                    End If
                    txt_Lsegmenta.Clear()
                    txt_Dsegmenta.Clear()
                    txt_Lsegmenta.Enabled = False
                    txt_Dsegmenta.Enabled = False
                    combo_lijevi.Enabled = False
                    combo_desni.Enabled = False
                    check_pero.Enabled = False
                    check_uskocnik.Enabled = False
                    btn_dodaj.Text = "Dodaj"
                    btn_izmjena.Text = "Izmjeni"
                    btn_brisi.Enabled = True
                    txt_index.Clear()
                    combo_lijevi.Text = ("")
                    combo_desni.Text = ("")
                    check_pero.Checked = False
                    check_uskocnik.Checked = False
                Catch ex As Exception
                    If txt1 = 0 Then
                        txt_Lsegmenta.BackColor = Color.LightCoral 'mjenja
boju polju sa krivim unosom
                        txt_Lsegmenta.Focus() 'stavlja fokus na polje sa
krivim unosom
                        PictureBox1.Show()
                        PictureBox2.Hide()
                    ElseIf txt2 = 0 Then
                        txt_Dsegmenta.BackColor = Color.LightCoral
                        txt_Dsegmenta.Focus()
                        PictureBox1.Hide()
                        PictureBox2.Show()
                    End If
                    MessageBox.Show("Molim unesite decimalni broj",
"Nepravilan unos.")
                End Try
            End If
        End If
    End Sub
```

```
    'funkcija za unos podataka u ListView. Napomenimo jos samo da svaka
lista ima vodecu velicinu (Item), ona sa indexom 0 te ostale pratece
(Subitems).
    Private Sub AddItemToListView()
        Dim lv As ListViewItem = List_all.Items.Add(List_all.Items.Count +
1)
        lv.SubItems.Add("true")
        lv.SubItems.Add(txt_Lsegmenta.Text)
        lv.SubItems.Add(txt_Dsegmenta.Text)
        If combo_lijevi.Text = "Radijus" Then
            lv.SubItems.Add("true")
            lv.SubItems.Add(r1)
        Else
            lv.SubItems.Add("false")
            lv.SubItems.Add("0")
        End If
        If combo_lijevi.Text = "Skošenje" Then
            lv.SubItems.Add("true")
            lv.SubItems.Add(S1_L)
            lv.SubItems.Add(S2_L)
        Else
            lv.SubItems.Add("false")
            lv.SubItems.Add("0")
            lv.SubItems.Add("0")
        End If
        If combo_desni.Text = "Radijus" Then
            lv.SubItems.Add("true")
            lv.SubItems.Add(r2)
        Else
            lv.SubItems.Add("false")
            lv.SubItems.Add("0")
        End If
        If combo_desni.Text = "Skošenje" Then
            lv.SubItems.Add("true")
            lv.SubItems.Add(S1_D)
            lv.SubItems.Add(S2_D)
        Else
            lv.SubItems.Add("false")
            lv.SubItems.Add("0")
            lv.SubItems.Add("0")
        End If
        If check_pero.Checked Then
            DimenzijePera()
            lv.SubItems.Add("true")
            lv.SubItems.Add(pero_form.A)
            lv.SubItems.Add(pero_form.B)
            lv.SubItems.Add(b_p)
            lv.SubItems.Add(t_p)
        Else
            lv.SubItems.Add("false")
            lv.SubItems.Add("0")
            lv.SubItems.Add("0")
            lv.SubItems.Add("0")
            lv.SubItems.Add("0")
        End If
        If check_uskocnik.Checked Then
            DimenzijeUskocnika()
            lv.SubItems.Add("true")
            lv.SubItems.Add(uskocnik_form.C)
            lv.SubItems.Add(e1)
            lv.SubItems.Add(CInt(txt_Dsegmenta.Text) - d)
```

```
        Else
            lv.SubItems.Add("false")
            lv.SubItems.Add("0")
            lv.SubItems.Add("0")
            lv.SubItems.Add("0")
        End If
    End Sub
    'funkcija za izmjenu podataka u ListView-u
    Private Sub EditItemInListView()
        If List_all.SelectedItems.Count > 0 Then
            List_all.SelectedItems(0).Text = txt_index.Text
            List_all.SelectedItems(0).SubItems(1).Text = ("true")
            List_all.SelectedItems(0).SubItems(2).Text = txt_Lsegmenta.Text
            List_all.SelectedItems(0).SubItems(3).Text = txt_Dsegmenta.Text
            If combo_lijevi.Text = "Radijus" Then
                List_all.SelectedItems(0).SubItems(4).Text = ("true")
                List_all.SelectedItems(0).SubItems(5).Text = r1
            Else
                List_all.SelectedItems(0).SubItems(4).Text = ("false")
                List_all.SelectedItems(0).SubItems(5).Text = ("0")
            End If
            If combo_lijevi.Text = "Skošenje" Then
                List_all.SelectedItems(0).SubItems(6).Text = ("true")
                List_all.SelectedItems(0).SubItems(7).Text = (S1_L)
                List_all.SelectedItems(0).SubItems(8).Text = (S2_L)
            Else
                List_all.SelectedItems(0).SubItems(6).Text = ("false")
                List_all.SelectedItems(0).SubItems(7).Text = ("0")
                List_all.SelectedItems(0).SubItems(8).Text = ("0")
            End If
            If combo_desni.Text = "Radijus" Then
                List_all.SelectedItems(0).SubItems(9).Text = ("true")
                List_all.SelectedItems(0).SubItems(10).Text = r2
            Else
                List_all.SelectedItems(0).SubItems(9).Text = ("false")
                List_all.SelectedItems(0).SubItems(10).Text = ("0")
            End If
            If combo_lijevi.Text = "Skošenje" Then
                List_all.SelectedItems(0).SubItems(11).Text = ("true")
                List_all.SelectedItems(0).SubItems(12).Text = (S1_D)
                List_all.SelectedItems(0).SubItems(13).Text = (S2_D)
            Else
                List_all.SelectedItems(0).SubItems(11).Text = ("false")
                List_all.SelectedItems(0).SubItems(12).Text = ("0")
                List_all.SelectedItems(0).SubItems(13).Text = ("0")
            End If
            If check_pero.Checked Then
                DimenzijePera()
                List_all.SelectedItems(0).SubItems(14).Text = ("true")
                List_all.SelectedItems(0).SubItems(15).Text = (pero_form.A)
                List_all.SelectedItems(0).SubItems(16).Text = (pero_form.B)
                List_all.SelectedItems(0).SubItems(17).Text = (b_p)
                List_all.SelectedItems(0).SubItems(18).Text = (t_p)
            Else
                List_all.SelectedItems(0).SubItems(14).Text = ("false")
                List_all.SelectedItems(0).SubItems(15).Text = ("0")
                List_all.SelectedItems(0).SubItems(16).Text = ("0")
                List_all.SelectedItems(0).SubItems(17).Text = ("0")
                List_all.SelectedItems(0).SubItems(18).Text = ("0")
            End If
            If check_uskocnik.Checked Then
```

```
                DimenzijeUskocnika()
                List_all.SelectedItems(0).SubItems(19).Text = ("true")
                List_all.SelectedItems(0).SubItems(20).Text =
(uskocnik_form.C)
                List_all.SelectedItems(0).SubItems(21).Text = (e1)
                List_all.SelectedItems(0).SubItems(22).Text =
((CInt(txt_Dsegmenta.Text) - d) / 2)
            Else
                List_all.SelectedItems(0).SubItems(19).Text = ("false")
                List_all.SelectedItems(0).SubItems(20).Text = ("0")
                List_all.SelectedItems(0).SubItems(21).Text = ("0")
                List_all.SelectedItems(0).SubItems(22).Text = ("0")
            End If
        End If
    End Sub
    'funkcija koja se odvija na pritisak Buttona "btn_izmjena"
    Private Sub btn_izmjena_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_izmjena.Click
        If txt_index.Text.Length > 0 Then 'If naredbom se osiguravamo da je
odabran redak za izmjenu
            If btn_izmjena.Text.ToLower() = "izmjeni" Then 'Prvi dio druge
If naredbe se odvija u slucaju da smo odabrali Button "btn_izmjena" kao
Izmjeni
                txt_Lsegmenta.Enabled = True
                txt_Dsegmenta.Enabled = True
                combo_lijevi.Enabled = True
                combo_desni.Enabled = True
                check_pero.Enabled = True
                check_uskocnik.Enabled = True
                btn_dodaj.Text = "Spremi"
                btn_izmjena.Text = "Odustani"
                btn_brisi.Enabled = False
                blnAdd = False
            Else 'Drugi dio se odvija u slucaju da smo odabrali Button
"btn_izmjena" kao Odustani
                txt_Lsegmenta.BackColor = Color.White
                txt_Dsegmenta.BackColor = Color.White
                txt_Lsegmenta.Enabled = False
                txt_Dsegmenta.Enabled = False
                combo_lijevi.Enabled = False
                combo_desni.Enabled = False
                check_pero.Enabled = False
                check_uskocnik.Enabled = False
                btn_dodaj.Text = "Dodaj"
                btn_izmjena.Text = "Izmjeni"
                btn_brisi.Enabled = True
                txt_index.Clear()
                txt_Lsegmenta.Clear()
                txt_Dsegmenta.Clear()
                combo_lijevi.Text = ("")
                combo_desni.Text = ("")
                check_pero.Checked = False
                check_uskocnik.Checked = False
            End If
        Else
            MessageBox.Show("Molim odaberite segment za izmjenu")
        End If
    End Sub
    'funkcija "btnbrisi_Click" brise odabrani redak u listi
    Private Sub btnbrisi_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_brisi.Click
```

```
        'za brisanje je poterbno odabrati redak te potvrditi samo brisanje
u MessageBox-u
        If List_all.SelectedItems.Count > 0 AndAlso MessageBox.Show("Jeste
li sigurni da želite obrisati ovaj segment?", "Potvrdi",
MessageBoxButtons.YesNo) = Windows.Forms.DialogResult.Yes Then
            Dim i As Integer 'brojac za for petlju u sljedecem retku
            For i = List_all.SelectedItems(0).Index To List_all.Items.Count
- 1 'For petlja sluzi za reguliranje indexa ostalim recima
                List_all.Items.Item(i).Text = i
            Next
            List_all.SelectedItems(0).Remove()
        End If
    End Sub
    'list_all_SelectedIndexChanged je funkcija koja preslikava podatke iz
odabranog reda liste u referentna polja
    Private Sub list_all_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
List_all.SelectedIndexChanged
        If List_all.SelectedItems.Count > 0 Then
            txt_index.Text = List_all.SelectedItems(0).Text
            txt_Lsegmenta.Text = List_all.SelectedItems(0).SubItems(2).Text
            txt_Dsegmenta.Text = List_all.SelectedItems(0).SubItems(3).Text
        End If
    End Sub
    'funkcije "DimenzijePera" te " "DimenzijeUskocnika" su tablice u kojima
se nalaze rukom uneseni podaci podaci.
    Private Sub DimenzijePera()
        Dim x As Double
        x = Convert.ToDouble(txt_Dsegmenta.Text)
        If 6 < x And x <= 8 Then
            b_p = 2
            t_p = 1.1
        ElseIf 8 < x And x <= 10 Then
            b_p = 3
            t_p = 1.7
        ElseIf 10 < x And x <= 12 Then
            b_p = 4
            t_p = 2.4
        ElseIf 12 < x And x <= 17 Then
            b_p = 5
            t_p = 2.9
        ElseIf 17 < x And x <= 22 Then
            b_p = 6
            t_p = 3.5
        ElseIf 22 < x And x <= 30 Then
            b_p = 8
            t_p = 4.1
        ElseIf 30 < x And x <= 38 Then
            b_p = 10
            t_p = 4.7
        ElseIf 38 < x And x <= 44 Then
            b_p = 12
            t_p = 4.9
        ElseIf 44 < x And x <= 50 Then
            b_p = 14
            t_p = 5.5
        ElseIf 50 < x And x <= 58 Then
            b_p = 16
            t_p = 6.2
        ElseIf 58 < x And x <= 65 Then
            b_p = 18
```

```
            t_p = 6.8
        ElseIf 65 < x And x <= 75 Then
            b_p = 20
            t_p = 7.4
        ElseIf 75 < x And x <= 85 Then
            b_p = 22
            t_p = 8.5
        ElseIf 85 < x And x <= 95 Then
            b_p = 25
            t_p = 8.7
        ElseIf 95 < x And x <= 110 Then
            b_p = 28
            t_p = 9.9
        ElseIf 110 < x And x <= 130 Then
            b_p = 32
            t_p = 11.1
        ElseIf 130 < x And x <= 150 Then
            b_p = 36
            t_p = 12.3
        ElseIf 150 < x And x <= 170 Then
            b_p = 40
            t_p = 13.5
        ElseIf 170 < x And x <= 200 Then
            b_p = 45
            t_p = 15.3
        ElseIf 200 < x And x <= 230 Then
            b_p = 50
            t_p = 17
        ElseIf 230 < x And x <= 260 Then
            b_p = 56
            t_p = 19.3
        ElseIf 260 < x And x <= 290 Then
            b_p = 63
            t_p = 19.6
        ElseIf 290 < x And x <= 330 Then
            b_p = 70
            t_p = 22
        ElseIf 330 < x And x <= 380 Then
            b_p = 80
            t_p = 24.6
        ElseIf 380 < x And x <= 440 Then
            b_p = 90
            t_p = 27.5
        ElseIf 440 < x And x <= 500 Then
            b_p = 100
            t_p = 30.4
        End If
    End Sub
    Private Sub DimenzijeUskocnika()
        Dim x As Double
        x = Convert.ToDouble(txt_Dsegmenta.Text)
        If x = 5 Then
            d = 4.8
            e1 = 0.7
        ElseIf x = 10 Then
            d = 9.6
            e1 = 1.1
        ElseIf x = 15 Then
            d = 14.3
            e1 = 1.1
        ElseIf x = 20 Then
```

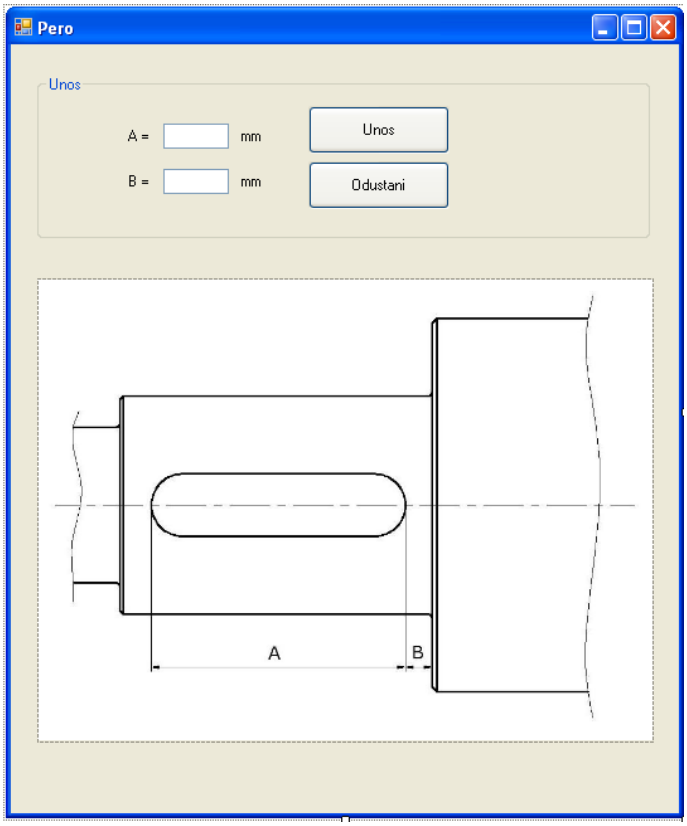
```
        d = 19
        e1 = 1.3
    ElseIf x = 25 Then
        d = 23.9
        e1 = 1.3
    ElseIf x = 30 Then
        d = 28.6
        e1 = 1.6
    ElseIf x = 35 Then
        d = 33
        e1 = 1.6
    ElseIf x = 40 Then
        d = 37.5
        e1 = 1.85
    ElseIf x = 45 Then
        d = 42.5
        e1 = 1.85
    ElseIf x = 50 Then
        d = 47
        e1 = 2.15
    ElseIf x = 55 Then
        d = 52
        e1 = 2.15
    ElseIf x = 60 Then
        d = 57
        e1 = 2.15
    ElseIf x = 65 Then
        d = 62
        e1 = 2.65
    ElseIf x = 70 Then
        d = 67
        e1 = 2.65
    ElseIf x = 75 Then
        d = 72
        e1 = 2.65
    ElseIf x = 80 Then
        d = 76.5
        e1 = 2.65
    ElseIf x = 85 Then
        d = 81.5
        e1 = 3.15
    ElseIf x = 90 Then
        d = 86.5
        e1 = 3.15
    ElseIf x = 95 Then
        d = 91.5
        e1 = 3.15
    ElseIf x = 100 Then
        d = 96.5
        e1 = 3.15
    ElseIf x = 105 Then
        d = 101
        e1 = 4.15
    ElseIf x = 110 Then
        d = 106
        e1 = 4.15
    ElseIf x = 115 Then
        d = 111
        e1 = 4.15
    ElseIf x = 120 Then
        d = 116
```

```
            e1 = 4.15
        ElseIf x = 125 Then
            d = 121
            e1 = 4.15
        ElseIf x = 130 Then
            d = 126
            e1 = 4.15
        End If

    End Sub
    'btn_zatvori_Click nam sluzi za takozvano zakljucavanje liste.
Usljucaju da nismo definirali svih 20 segmenata program automatski ispuni
ostale segmente sa defaultnim vrijednostima.
    Private Sub btn_zatvori_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_zatvori.Click
        Dim i As Integer 'brojac za for petlju
        If btn_zatvori.Text.ToLower() = "zatvori listu" Then 'zakjucavanje
liste
            ProvjeraPodataka() 'poziv na funkciju ProvjeraPodataka
            If List_all.Items.Count <> 20 Then
                For i = List_all.Items.Count + 1 To 20
                    Dim lv As ListViewItem =
List_all.Items.Add(List_all.Items.Count + 1)
                    lv.SubItems.Add("false")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("false")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("false")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("false")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("false")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("false")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("false")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("0")
                    lv.SubItems.Add("0")
                Next
            End If
            btn_zatvori.Text() = "Otvori listu"
            btn_brisi.Enabled = False
            btn_dodaj.Enabled = False
            btn_izmjena.Enabled = False
            btn_export.Enabled = True
        Else 'otkljucavanje liste u slucaju potrebe za promjenama
            For i = List_all.Items.Count - 1 To 0 Step -1
                If List_all.Items(i).SubItems(1).Text = "false" Then
                    List_all.Items(i).Remove() 'program odstranjuje iz
liste sve clanove koji definiraju neaktivne segmente
                End If
            Next
            btn_zatvori.Text() = "Zatvori listu"
            btn_brisi.Enabled = True
```

```
            btn_dodaj.Enabled = True
            btn_izmjena.Enabled = True
            btn_export.Enabled = False
        End If
    End Sub
    Private Sub ProvjeraPodataka() 'Funkcija koja provjerava tocnost
unesenih podataka.
        'Sprijecava greske poput takve da duljina pera bude veca od samog
segmenta i slicno.
        Dim i As Integer
        For i = 0 To List_all.Items.Count - 1
            Dim duljina, ref1, ref2 As Double
            duljina = Convert.ToDouble(List_all.Items(i).SubItems(2).Text)
            ref1 = Convert.ToDouble(List_all.Items(i).SubItems(15).Text) +
Convert.ToDouble(List_all.Items(i).SubItems(16).Text)
            ref2 = Convert.ToDouble(List_all.Items(i).SubItems(20).Text)
            If ref1 >= duljina Then
                'ref1 - zbroj duljine utora za pero i udaljenosti pocetka
otora od susjednog segmenta
                'duljina - duljina segmenta
                'u slucaju greske javlja se MessageBox koji nas opominje na
istu te nas vodi u novi unos podataka o utoru za pero
                txt_index.Text = List_all.Items(i).Text
                txt_Lsegmenta.Text = List_all.Items(i).SubItems(2).Text
                txt_Dsegmenta.Text = List_all.Items(i).SubItems(3).Text
                MessageBox.Show("Utor za pero na segmentu broj " & i + 1 &
" nije pravilno definiran. Molim promijenite unos.", "Greška")
                pero_form.btn_Odustani.Visible = False
                pero_form.ShowDialog()
                DimenzijePera()
                List_all.Items(i).SubItems(15).Text = (pero_form.A)
                List_all.Items(i).SubItems(16).Text = (pero_form.B)
                List_all.Items(i).SubItems(17).Text = (b_p)
                List_all.Items(i).SubItems(18).Text = (t_p)
            End If
            If ref2 >= duljina Then
                'ref2 - udaljenost utora za uskocnik od ruba segmenta
                'duljina - duljina segmenta
                'u slucaju greske javlja se MessageBox koji nas opominje na
istu te nas vodi u novi unos podataka o utoru za uskocnik
                txt_index.Text = List_all.Items(i).Text
                txt_index.Text = List_all.Items(i).Text
                txt_Lsegmenta.Text = List_all.Items(i).SubItems(2).Text
                txt_Dsegmenta.Text = List_all.Items(i).SubItems(3).Text
                MessageBox.Show("Utor za uskočnik na segmentu broj " & i +
1 & " nije pravilno definiran. Molim promijenite unos.", "Greška")
                uskocnik_form.btn_Odustani.Visible = False
                uskocnik_form.ShowDialog()
                DimenzijeUskocnika()
                List_all.Items(i).SubItems(20).Text = (uskocnik_form.C)
                List_all.Items(i).SubItems(21).Text = (d)
                List_all.Items(i).SubItems(22).Text = (e1)
            End If
        Next
    End Sub
    Private Sub btn_export_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_export.Click
        Dim a(1000) As String 'jednodimenzionalno polje sa podacima iz
liste
        Dim i, j, n As Integer
```

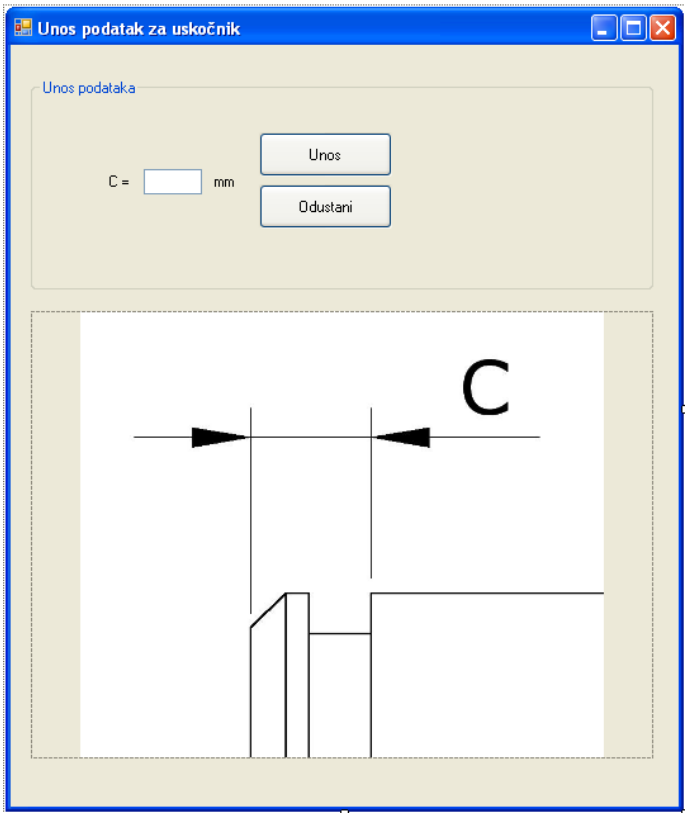
```
        'for petlje sluze za preoblikovanje liste u oblik potreban za xls
datoteku
        'od 2D liste dobivamo 1D
        For i = 1 To List_all.Columns.Count() - 1
            For j = 0 To List_all.Items.Count() - 1
                a(n) = List_all.Items(j).SubItems(i).Text
                If i = 3 Then a(n) = a(n) / 2
                n = n + 1
            Next
        Next
        Dim oExcel As Microsoft.Office.Interop.Excel.Application = New
Microsoft.Office.Interop.Excel.Application
        Dim oWorkBook As Microsoft.Office.Interop.Excel.Workbook
        Dim oWorkSheet As Microsoft.Office.Interop.Excel.Worksheet
        Dim fileD As New OpenFileDialog()
        fileD.ShowDialog()
        Dim fileName As String = fileD.FileName
        oWorkBook = oExcel.Workbooks.Open(fileName)
        oWorkSheet = oWorkBook.Worksheets().Item(1)
        oWorkSheet.Activate()
        For i = 1 To List_all.Items.Count * 22 'petlja koja sprema podatke
u celije odabrane .xls datoteke
            oWorkSheet.Cells(i, 3).Value = a(i - 1)
        Next
        oWorkBook.Application.DisplayAlerts = False
        oWorkBook.SaveAs(fileName) 'sprema datoteku
        oWorkBook.Close()
        Try
            TryToDisplayGeneratedFile(fileName)
        Catch
            MsgBox("Molim zatvorite sve aktivne Excel datoteke")
        End Try
    End Sub
    Sub TryToDisplayGeneratedFile(ByVal fileName As String)
        Try
            System.Diagnostics.Process.Start(fileName)
        Catch
            Console.WriteLine(fileName + " created in application folder.")
        End Try
    End Sub
End Class
```

Sučelje za unos podataka koji opisuju utor za pero



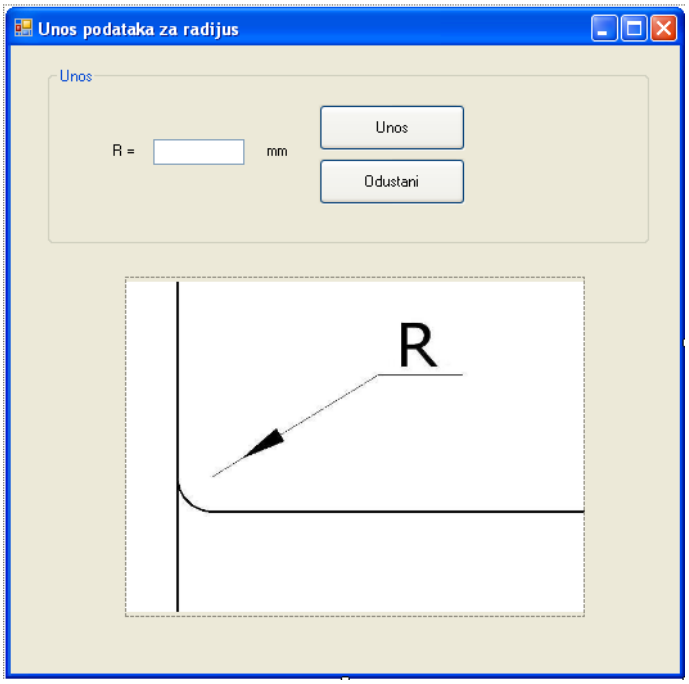
```
'kod od forme "Pero_form" koji se aktivira ukljucivanjem CheckBox-a pero
ili kod potrebe za izmjenom podataka koji definiraju pero
Public Class Pero_form
    Public Shared A, B As String 'public shared varijable su dostupne svim
formama u projektu
    Private Sub txt_A_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txt_A.TextChanged
        'prilikom izmjene teksta u polju, varijabla odmah poprima
vrijednost unesenog teksta
        A = txt_A.Text
        txt_A.BackColor = Color.White 'pozadina polja poprima bijelu boju
    End Sub
    Private Sub txt_B_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txt_B.TextChanged
        B = txt_B.Text
        txt_B.BackColor = Color.White
    End Sub
    'pritiskom na btn_unos program prvo provjerava dali su uneseni
decimalni brojevi
    'u slucaju da jesu, forma sprema iznose u varijable te se zatvara
    'u protivnom se javlja poruka "Molim unesite decimalni broj"
    Private Sub btn_unos_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_unos.Click
        Dim brojA, brojB As Double
        Try
            brojA = Convert.ToDouble(txt_A.Text)
            brojB = Convert.ToDouble(txt_B.Text)
            btn_Odustani.Visible = True
            Me.Hide()
        Catch ex As Exception
            'ako se pojavila greška u unosu podataka u određeno polje isto
polje mijenja boju pozadine
            If brojA = 0 Then
                txt_A.BackColor = Color.LightCoral
            ElseIf brojB = 0 Then
                txt_B.BackColor = Color.LightCoral
            End If
            MessageBox.Show("Molim unesite decimalni broj", "Nepravilan
unos")
        End Try
    End Sub
    'pritiskom na Button "btn_Odustani" forma se zatvara bez dodavanja
ikakvih velicina varijablama
    Private Sub btn_Odustani_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btn_Odustani.Click
        Me.Hide()
        Glavna.check_pero.Checked = False
    End Sub
    'funkcija koja sluzi za ciscenje svih polja prilikom novog pokretanja
forme i mijenjanje pozadine na osnovnu bijelu
    Private Sub Pero_form_Shown(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Shown
        txt_A.Clear()
        txt_B.Clear()
        txt_A.BackColor = Color.White
        txt_B.BackColor = Color.White
    End Sub
      End Class
```

Sučelje za unos podataka koji opisuju utor za uskočnik



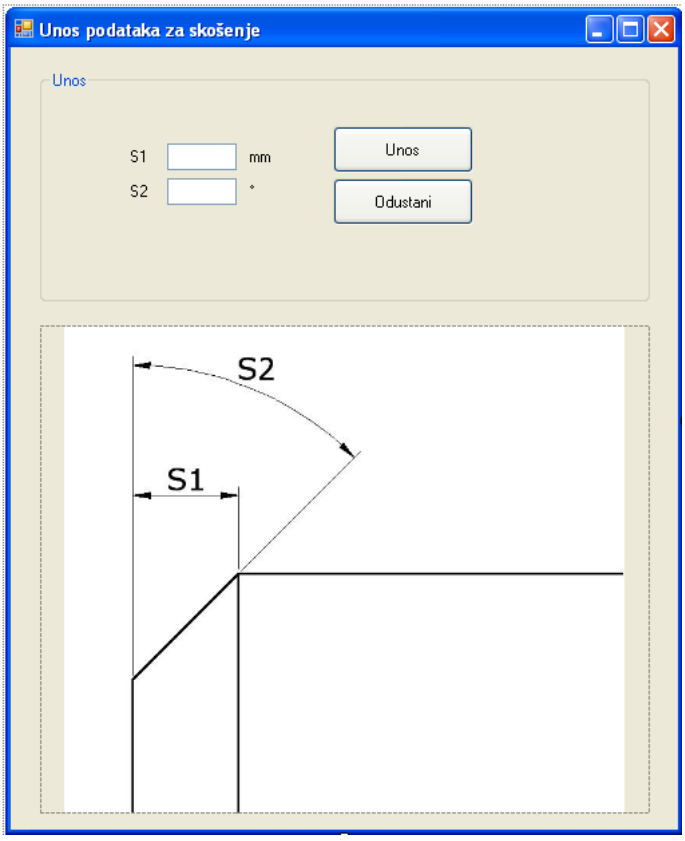
```
'kod koji se odvija aktiviranjem forme "Uskocnik_form" iste je strukture
kao i kod forme "Pero_form"
'iz tog razloga smatram da nije potrebno pisat ponovno iste komentare
Public Class Uskocnik_form
    Public Shared C As String
    Private Sub txt_C_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txt_C.TextChanged
        txt_C.BackColor = Color.White
        C = txt_C.Text
    End Sub
    Private Sub Unos_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_Unos.Click
        Dim brojC As Double
        Try
            brojC = Convert.ToDouble(txt_C.Text)
            Me.Hide()

        Catch ex As Exception
            txt_C.BackColor = Color.LightCoral
            MessageBox.Show("Molim unesite decimalni broj", "Nepravilan
unos")
        End Try
    End Sub
    Private Sub Odustani_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btn_Odustani.Click
        Me.Hide()
        Glavna.check_uskocnik.Checked = False
    End Sub
    Private Sub Uskocnik_form_Shown(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Shown
        txt_C.Clear()
        txt_C.BackColor = Color.White
    End Sub
      End Class
```

Sučelje za unos radijusa zaobljenja na kraju segmenta



```
Public Class Radijus
    'pokrece se aktiviranjem forme Radijus
    Public Shared R As String
    Public Shared radlijevi, raddesni As Boolean 'boolean varijable
definiraju koji se radijus unosi (lijevi/desni)
    Private Sub txtRadijus_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles txtRadijus.TextChanged
        R = txtRadijus.Text 'unos radijusa
        txtRadijus.BackColor = Color.White
    End Sub
    Private Sub Unos_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Unos.Click
        Dim brojR As Double
        Try
            brojR = CDbl(txtRadijus.Text) 'transformacija radijusa iz
string u double
            Me.Hide()
            pbox_desna.Hide()
            pbox_lijeva.Hide()
        Catch ex As Exception
            txtRadijus.BackColor = Color.LightCoral
            MessageBox.Show("Molim unesite decimalni broj", "Nepravilan
unos")
        End Try
    End Sub
    Private Sub Odustani_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Odustani.Click
        pbox_desna.Hide()
        pbox_lijeva.Hide()
        Me.Hide()
        'naredne If petlje reguliraju koji ce se ComboBox resetirat u
slucaju odustajanja od unosa
        If radlijevi = True Then
            Glavna.combo_lijevi.Text =
Glavna.combo_lijevi.Items.Item(0).ToString
        End If
        If raddesni = True Then
            Glavna.combo_desni.Text =
Glavna.combo_desni.Items.Item(0).ToString
        End If
    End Sub
    Private Sub Radijus_Shown(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Shown
        txtRadijus.Clear()
        txtRadijus.BackColor = Color.White
    End Sub
      End Class
```

Sučelje za unos dimenzija skošenja na kraju segmenta



```
Public Class Skosenje
    Public Shared S1, S2 As String
    Public Shared skolijevo, skodesno As Boolean
    Private Sub txt_S1_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles txt_S1.TextChanged
        S1 = txt_S1.Text
        txt_S1.BackColor = Color.White
    End Sub
    Private Sub txt_S2_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles txt_S2.TextChanged
        S2 = txt_S2.Text
        txt_S2.BackColor = Color.White
    End Sub
    Private Sub Unos_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Unos.Click
        Dim brojS1, brojS2 As Double
        Try
            brojS1 = Convert.ToDouble(txt_S1.Text)
            brojS2 = Convert.ToDouble(txt_S2.Text)
            pbox_lijeva.Hide()
            pbox_desna.Hide()
            Me.Hide()
        Catch ex As Exception
            If brojS1 = 0 Then
                txt_S1.BackColor = Color.LightCoral
            ElseIf brojS2 = 0 Then
                txt_S2.BackColor = Color.LightCoral
            End If
            MessageBox.Show("Molim unesite decimalni broj", "Nepravilan
unos")
        End Try
    End Sub
    Private Sub Odustani_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Odustani.Click
        pbox_lijeva.Hide()
        pbox_desna.Hide()
        Me.Hide()
        'naredne If petlje reguliraju koji ce se ComboBox resetirat u
slucaju odustajanja od unosa
        If skolijevo = True Then
            Glavna.combo_lijevi.Text =
Glavna.combo_lijevi.Items.Item(0).ToString
        End If
        If skodesno = True Then
            Glavna.combo_desni.Text =
Glavna.combo_desni.Items.Item(0).ToString
        End If
    End Sub
    Private Sub Skosenje_Shown(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Shown
        txt_S1.Clear()
        txt_S2.Clear()
        txt_S1.BackColor = Color.White
        txt_S2.BackColor = Color.White
    End Sub
      End Class
```