

### 3. Grafičke kontrole

Koristeći programe iz paketa Microsoft Office, npr. MS Word, oblikovali ste tekst klikajući na slikovne gumbе – za podebljavanje teksta kliknuli ste na **B** a za podcrtavanje na **U**. Ako ste željeli promijeniti veličinu pisma upisali ste to u okvir .



Slika 50. Grafičke kontrole

Sve su to grafičke kontrole pomoću kojih ste oblikovali tekst. Iza svake te grafičke kontrole stoje naredbe koje oblikuju tekst upravo onako kako ste vi to zadali. Sada ćete upoznati kako programirati neke grafičke kontrole. Na slici 50. označeni su:

- 1 prozor u kojem se nalaze grafičke kontrole
  - 2 okvir za upis teksta, **Entry**
  - 3 okvir poruke, **Label** - služe za ispis teksta u prozoru npr. metara ili ispis rezultata računске operacije – npr. *1600*.
  - 4 naredbeni gumb, **Button**
- Upoznajmo ih redom rješavajući nekoliko vježbi.

► Programiranje grafičkih kontrola omogućuje modul `tkinter`, pa na početku programa u kojem koristimo grafičke kontrole treba unijeti taj modul naredbom:

```
from tkinter import *
```

`tkinter` – modul koji sadrži grafičke kontrole

#### Vježba 16.

Napišite program koji zadaje dva naredbena gumba:

- Kvadrat – crta kvadrat stranice duljine 100 nasumično odabrane boje
- Desno – okreće kornjaču udesno za  $30^\circ$ . Pomoću ta dva gumba nacrtati dvanaest kvadrata sa zajedničkom točkom.

#### Rješenje:

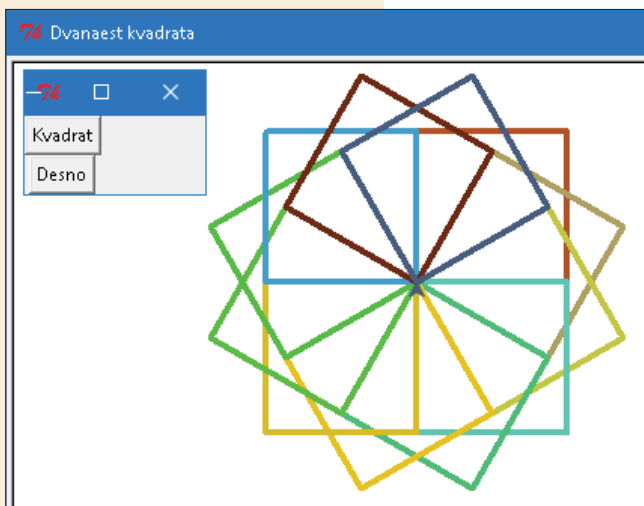
Promotrite program:

Potprogrami su vam već poznati i znate čemu služe:

- `kvadrat()` – crta kvadrat stranice duljine 100
- `desno()` – okreće kornjaču udesno za  $30^\circ$
- `boja()` – zadaje nasumično određenu boju kornjače (koristi se u potprogramu `kvadrat()`)

Promotrimo redom naredbe glavnog programa:

1. Zadavanje prozora za naredbene gumbе: `prozor = Tk()`, gdje je prozor ime prozora.
2. Neka je naslov prozora `Gumbi`: `prozor.title('Gumbi')`



Slika 51. Dvanaest kvadrata sa zajedničkom točkom

```

from turtle import *
from random import *
from tkinter import *
def kvadrat():
    boja()
    for k in range(4):
        fd(100); rt(90)

def desno():
    rt(30)

def boja():
    c = randrange(0,255)
    z = randrange(0,255)
    p = randrange(0,255)
    color(c,z,p)

prozor = Tk()
prozor.title('Gumbi')
title('Dvanaest kvadrata')
colormode(255); st(); lt(90); width(4)
glavni = Frame(prozor)
glavni.grid(columns = 1, rows = 2)
g_kvadrat = Button(glavni, text = 'Kvadrat', command = kvadrat)
g_kvadrat.grid(column = 1, row = 1)
g_desno = Button(glavni, text = 'Desno', command = desno)
g_desno.grid(column = 1, row = 2)

```

3. Zadaje se naslov prozora u kojem crta kornjača, namještamo oblik boje, usmjerenje i debljinu pera kornjače: `title('Dvanaest kvadrata');` `colormode(255); st(); lt(90); width(4)`

4. U prozor za naredbene gumbе stavlja se okvir (engl. *frame* – okvir) u koji će doći gumbi, ime okvira je glavni: `glavni = Frame(prozor)`. Zadajemo svojstva okvira glavni: `glavni.grid(columns = 1, rows = 2)`. Ovom naredbom prozor dijelimo na “police” u koje ćemo stavlјati gumbе. Zadali smo dva reda i jedan stupac “polica”. Ove dvije naredbe uvijek dolaze zajedno, prije zadavanja grafičkih kontrola.

5. Sada zadajete naredbeni gumb (engl. *button* – gumb) za crtanje kvadrata naredbama:

```

g_kvadrat = Button(glavni, text = 'Kvadrat', command
    = kvadrat)
g_kvadrat.grid(column = 1, row = 1)

```

Varijabli `g_kvadrat` pridružujete naredbeni gumb `Button` s ulaznim vrijednostima:

- ime prozora u kojem se kontrola nalazi: `glavni`
- tekst koji će biti ispisан na gumbu: `text = 'Kvadrat'`
- ime procedure koju treba pozvati kad se klikne na gumb: `command = kvadrat`

Drugom naredbom zadajete položaj gumba svojstvom `.grid`: gumb za crtanje kvadrata nalazi se u 1. stupcu i 1. retku okvira. (engl. *column* – stupac, *row* – redak).

6. Slično zadajete naredbeni gumb za okretanje kornjače udesno naredbom:

```
g_desno = Button(glavni, text = 'Desno', command = desno)
g_desno.grid(column = 1, row = 2)
```

Sada je gumb za okret udesno u 1. stupcu, ali u 2. retku, pa je ispod gumba kvadrat.

 Rješenje u datoteci: **86Bvje16.py**.

### ► Prozor s grafičkim kontrolama

Zadavanje prozora za grafičke kontrole:

```
ime = Tk(), gdje je ime ime prozora.
```

Zadavanje naslova prozora ime: `ime.title('Naslov')`

Grafičke kontrole stavljamo u okvir unutar prozora.

Zadavanje okvira: `okvir = Frame(ime)`, gdje je *ime* ime prozora.

Svojstva okvira *okvir*: `okvir.grid(rows = 2, columns = 2)`

### ► Naredbeni gumb **Button**

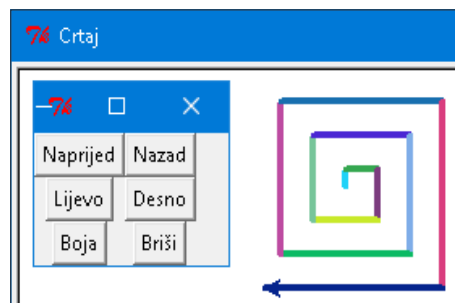
Osnovni oblik naredbe za zadavanje gumba je:

```
ime_gumba = Button(okvir, text = "opis", command = procedura)
ime_gumba.grid(column = stupac, row = red).
```

### Vježba 17.

Napišite program koji zadaje 6 naredbenih gumba, kao na slici 52., koji omogućuju crtanje pomoću kornjače. Treba zadati sljedeće gumbе:

- Naprijed – pomak naprijed za 10
- Nazad – pomak nazad za 10
- Lijevo – okret ulijevo za 90°
- Desno – okret udesno za 90°
- Boja – zadaje nasumičnu boju
- Briši – briše kornjačin prozor



Slika 52. Crtanje pomoću

### Rješenje:

Promotrite program:

```

from turtle import *
from random import *
from tkinter import *
def naprijed():
    fd(10)
def nazad():
    bk(10)
def desno():
    rt(90)
def lijevo():
    lt(90)
def briši():
    reset()
def boja():
    c = randrange(0,255)
    z = randrange(0,255)
    p = randrange(0,255)
    color(c,z,p)

prozor = Tk()
prozor.title('Gumbi')
title('Crtaj')
colormode(255); st(); lt(90); width(4)
glavni = Frame(prozor); glavni.grid(column = 2, row = 3)
g_naprijed = Button(glavni, text = 'Naprijed', command = naprijed)
g_naprijed.grid(column = 1, row = 1)
g_nazad = Button(glavni, text = 'Nazad', command = nazad)
g_nazad.grid(column = 2, row = 1)
g_lijevo = Button(glavni, text = 'Lijevo', command = lijevo)
g_lijevo.grid(column = 1, row = 2)
g_desno = Button(glavni, text = 'Desno', command = desno)
g_desno.grid(column = 2, row = 2)
g_boja = Button(glavni, text = 'Boja', command = boja)
g_boja.grid(column = 1, row = 3)
g_boja = Button(glavni, text = 'Briši', command = briši)
g_boja.grid(column = 2, row = 3)

```

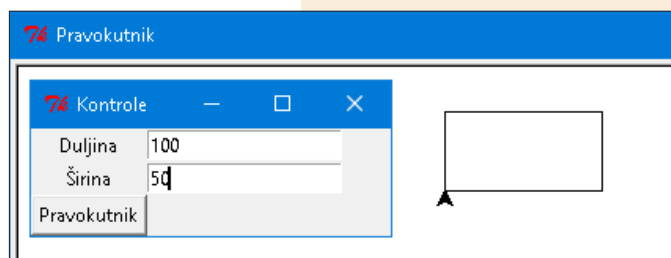


Rješenje u datoteci: **86Bvje17.py**.

### Vježba 18.

Napišite program koji u prozor s kontrolama stavlja sljedećih 5 kontrola:

- okvir poruke: Duljina
- okvir za upis duljine pravokutnika
- okvir poruke: Širina
- okvir za upis širine pravokutnika
- naredbeni gumb za crtanje pravokutnika zadane duljine i širine.



**Slika 53.** Crtanje pravokutnika zadane duljine i širine

Reci koji u programu počinju znakom # i ispisuju se crvenom bojom su komentari. Oni se ne izvode pa obično služe za opis pojedinog dijela programa.

Evo čitavog programa:

### Rješenje:

Kontrola za okvir poruke zove se Label . Naredba koja je ispisala riječ *Duljina* u prozoru s kontrolama je: `t1 = Label(glavni, text = 'Duljina')`

Njene osnovne ulazne vrijednosti su:

- `glavni` – ime okvira u kojem se nalaze kontrole
- `text = 'Duljina'` – tekst koji treba ispisati

Položaj kontrole u okviru zadali smo naredbom `t1.grid(column = 1, row = 1)`, kao i za Button kontrolu.

Kontrola za upis podatka zove se Entry (engl. *entry* – ulazak). Oblik naredbe je sljedeći:

```
duljina = Entry(glavni)
duljina.grid(column = 2, row = 1)
```

gdje je:

- `duljina` – ime varijable u kojoj će se pamtititi podatak koji upisujemo u kontrolu
- `glavni` - ime okvira u kojem se kontrola nalazi


```
from tkinter import *
from turtle import *
def pravokutnik():
    a = int(duljina.get())
    b = int(širina.get())
    for k in range(2):
        fd(b); rt(90)
        fd(a); rt(90)

prozor = Tk()
prozor.title('Kontrole')
title('Pravokutnik')
colormode(255); st(); lt(90)
glavni = Frame(prozor)
glavni.grid(column = 2, row = 2)
#upis duljine pravokutnika
t1 = Label(glavni, text = 'Duljina')
t1.grid(column = 1, row = 1)
duljina = Entry(glavni)
duljina.grid(column = 2, row = 1)
#upis širine pravokutnika
t2 = Label(glavni, text = 'Širina')
t2.grid(column = 1, row = 2)
širina = Entry(glavni)
širina.grid(column = 2, row = 2)
#gumb za crtanje pravokutnika
gumb = Button(glavni, text = 'Pravokutnik', command = pravokutnik)
gumb.grid(column = 1, row = 3)
```

- duljina.grid(column = 2, row = 1) - položaj kontrole u okviru, isto kao i u Button kontroli

Na isti način zadajte kontrolu za upis širine pravokutnika.

I napokon, nakon što kliknete na gumb za crtanje pravokutnika, potrebno je pročitati podatke koji se nalaze u Entry kontrolama. Duljinu pravokutnika pročitat ćete naredbom: `a = int(duljina.get())`, gdje `duljina.get()` daje tekstualni podatak upisan u kontrolu, pa ga naredbom `int` pretvaramo u cijeli broj.

 Rješenje u datoteci: **86Bvje18.py**.

Primijetite da svojstvo `.get` čita broj koji se upiše u okvir i vraća ga u obliku teksta.

▶ Grafička kontrola pomoću koje ispisujemo tekst u prozoru s kontrolama je **Label**. Oblik naredbe je:

```
ime = Label(okvir, text = 'poruka')
ime.grid(column = stupac, row = red)
```

Grafička kontrola koja služi za unos podataka je **Entry**. Oblik naredbe je:

```
ulaz = Entry(okvir)
ulaz.grid(column = 2, row = 1).
```

### Vježba 19.

Napišite program za pretvaranje metara u centimetre.

Program treba zadati sljedeće kontrole:

- jednu *Entry* kontrolu za unos metara
- jednu *Button* kontrolu za izračunavanje
- dvije *Label* kontrole za ispis sljedećih poruka:
  - "metara"
  - "iznosi centimetara" i iznosa centimetara.

### Rješenje:

Na slici 54. tekst je ispisan većim pismom. Zato je u kontroli *Label* dodano svojstvo za zadavanje fonta oblika `font = ('Arial', 12, 'bold')`.

Potprogram `pretvori()` pretvara metre u centimetre i ispisuje rezultat u *Label* kontroli:

```
def pretvori():
    broj = int(metri.get())
    centimetri.config(text = 'iznosi ' + str(broj *
    100) + ' centimetara')
```

Naredba `broj = int(metri.get())` u varijablu `broj` sprema upisani broj metara iz *Entry* kontrole.



Slika 54. Pretvaranje metara u centimetre

`str(n)` pretvara broj `n` u tekst

```

from tkinter import *
def pretvori():
    broj = int(metri.get())
    centimetri.config(text = 'iznosi ' + str(broj * 100) + ' centimetara')

prozor = Tk()
prozor.title('Metri u centimetre')
glavni = Frame(prozor)
glavni.grid(column = 3, row = 3)
metri = Entry(glavni, font = ('Arial',12,'bold'))
metri.grid(column = 2, row = 1)
#ispis teksta u prozoru
t = Label(glavni, text = 'metara', font = ('Arial', 12, 'bold'))
t.grid(column = 3, row = 1)
#okvir za ispis rezultata
centimetri = Label(glavni, font = ('Arial' ,12 ,'bold'))
centimetri.grid(column = 1, row = 2, columnspan = 2)
#gumb za pretvorbu
gumb = Button(glavni, font = ('Arial',12,'bold'),text = 'Pretvori', command = pretvori)
gumb.grid(column = 3, row = 3)

```

engl. *configure* – podesiti

▶ Naredba `.get()` vraća tekstualni podatak upisan u kontroli *Entry*.  
Naredba `.config` postavlja vrijednost svojstva `text` kontrole *Label*.

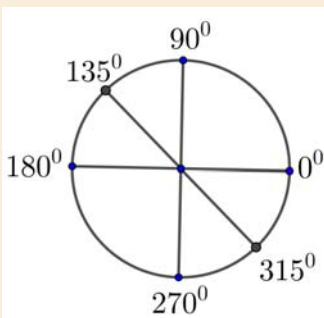
Druga naredba zapisuje rezultat u *Label* kontrolu *centimetri*. Vrijednost *broj* množimo sa 100 (1 m = 100 cm) i pretvaramo ga u tekst naredbom `str(broj * 100)`. Naredbom `.config` postavljamo vrijednost svojstva `text` – to je tekst koji će pisati u kontroli *Label*.

▶ Rješenje u datoteci: **86Bvje19.py**.

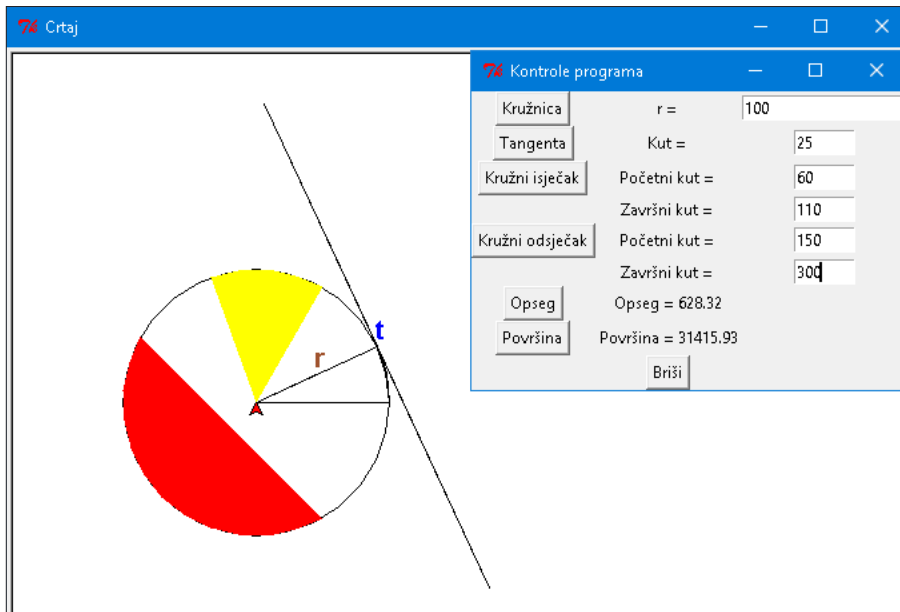
### Vježba 20.

Napišite program koji zadaje prozor s grafičkim kontrolama:

- **Kružnica** crta kružnicu čija duljina polumjera je zadana u okviru za upis teksta
- **Tangenta** crta tangentu u točki kružnice koja je zadana kutom, a kut se upisuje u okviru za upis teksta
- **Kružni isječak** crta kružni isječak obojan žutom bojom, čiji se početni i završni kut zadaje okvirom za upis teksta
- **Kružni odsječak** crta kružni isječak obojan crvenom bojom, čiji se početni i završni kut zadaje okvirom za upis teksta
- **Opseg** računa opseg kruga i ispisuje ga u okviru poruke pokraj gumba
- **Površina** računa površinu kruga ispisuje ju u okviru poruke pokraj gumba
- **Briši** briše grafički prozor



**Slika 56.** Svaka točka na kružnici zadana je kutom između  $0^\circ$  i  $360^\circ$



**Slika 55.** Crtanje kružnice, tangente, kružnog isječka, kružnog odsječka i računanje opsega i površine kruga pomoću grafičkih kontrola

Potprograme koji crtaju kružnicu, tangentu i kružni isječak već smo opisali. Sigurno ste već napisali i potprogram za crtanje kružnog odsječka. Ali, sada vrijednosti zadane okvirima za upis poruke bit će globalne varijable. Dakle, potprogrami neće imati ulazne vrijednosti. Napišimo prvo program koji daje grafičke kontrole za kružnicu i crta kružnicu:

- Treba unijeti modul za grafičke kontrole `tkinter` i modul `math` (zbog broja  $\pi$ )
- Potprogram `kružnica()` crta kružnicu čiji je polumjer zadan u okviru poruke `r`, a broj zapisan u tom okviru stavljate u varijablu `r1`. U globalnoj varijabli `x` pamti se središte kružnice. Naime, nakon crtanja isječka, tangente ili odsječka, najbolje je vratiti kornjaču u središte kružnice. Time se omogućuje da se gumbi mogu klikati bilo koji redom.
- Crtež se uobičajeno briše naredbom `reset()`. Ostaje samo kornjaču usmjeriti „prema gore“.

U glavnom programu zadajemo prozor za grafičke kontrole:

```
prozor = Tk()
prozor.title('Kontrole programa')
title('Crtaj') ; lt(90)
```

```
from tkinter import *
from math import *
def kružnica():
    global x
    r1 = int(r.get())
    circle(r1)
    lt(90); fd(r1); rt(90)
    x = pos()

def briši():
    reset()
    seth(90)
```



Komentari u Pythonu: počinju s # i crvene su boje.

Položaj kontrola u prozoru najbolje je prikazati tablicom:

redak	stupac		
	1.	2.	3.
1.	Kružnica	r =	<input type="text"/>
2.	Tangenta	Kut =	<input type="text"/>
3.	Kružni isječak	Početni kut =	<input type="text"/>
4.		Završni kut =	<input type="text"/>
5.	Kružni odsječak	Početni kut =	<input type="text"/>
6.		Završni kut =	<input type="text"/>
7.	Opseg		
8.	Površina		
9.		Briši	

**Tablica 1.** Položaj grafičkih kontrola u prozoru Kontrole programa

### 1. Kružnica i brisanje crteža

Slijede kontrole za crtanje kružnice. Radi preglednosti, napisat ćemo: # kontrole: kružnica. Znak # označava Pythonu da iza slijedi tekst te da u tom redu nisu naredbe. Taj tekst služi nama za lakše praćenje naredbi u programu. Kažemo da je to komentar u programu.

Prvo zadajete gumb `Kružnica` i stavljate ga u 1. red i 1. stupac

```
gumb_kr = Button(prozor, text = 'Kružnica', command = kruznica)
gumb_kr.grid(column = 1, row = 1)
```

U drugi stupac istog reda stavljate okvir teksta `r =`:

```
r_opis = Label(prozor, text = 'r = ')
r_opis.grid(column = 2, row = 1)
```

U treći stupac dodajete okvir za upis teksta: :

```
r = Entry(prozor)
r.grid(column = 3, row = 1)
```

Dodajte još gumb `Briši`. Stavite ga odmah u drugi stupac devetog reda, makar će se on privremeno prikazati u drugom redu dok niste stavili kontrole u redove između 1. i 9. reda.

#### #brisanje crteža

```
brisi = Button(prozor, text = 'Briši', command=brisi)
brisi.grid(column=2, row=9)
```

Time ste sastavili glavni program koji za sada samo crta kružnicu i briše crtež. Pazite, početak programa s dva potprograma je u gornjem okviru:

```

prozor = Tk()
prozor.title('Kontrole programa')
title('Crtaj'); lt(90)

#kontrola: kruznica
gumb_kr = Button(prozor, text = 'Kružnica', command = kruznica)
gumb_kr.grid(column = 1, row = 1)
r_opis = Label(prozor, text = 'r = ')
r_opis.grid(column = 2, row = 1)
r = Entry(prozor)
r.grid(column = 3, row = 1)

#brisanje crteža
brisi = Button(prozor, text='Briši', command=briši)
brisi.grid(column=2, row=9)

```

## 2. Tangenta

Dopunimo sada program potprogramom tangenta i grafičkim kontrolama za tangentu:

Potprogram tangenta():

```

def tangenta():
    r1 = int(r.get())
    k = int(kutt.get())
    rt(90); fd(r1); lt(90)
    circle(r1,k)
    fd(2*r1); bk(4*r1); fd(2*r1)
    color('blue')
    write('t',font=('Arial',16,'bold')); pu();bk(30); pd()
    pu(); fd(30); pd(); color('black');lt(90); fd(r1); bk(r1/2)
    color('Sienna')
    write('r',font=('Arial',16,'bold')); pu();bk(30)

```

Glavni program treba dopuniti naredbama:

```

#kontrola: tangenta
gumb_tan = Button(prozor, text = 'Tangenta', command = tangenta)
gumb_tan.grid(column = 1, row = 2)
tan_opis = Label(prozor, text = 'Kut = ')
tan_opis.grid(column = 2, row = 2)
kutt = Entry(prozor, width = 7)
kutt.grid(column = 3, row = 2)

```

### 3. Kružni isječak

Potprogram za crtanje kružnog isječka:

```
def kružni_isjecak():
    r1 = int(r.get())
    kut1 = int(kutp.get())
    kut2 = int(kutz.get())
    color('black', 'yellow')
    begin_fill()
    seth(kut1);
    fd(r1); lt(90)
    circle(r1, kut2-kut1)
    lt(90); fd(r1)
    end_fill()
    seth(90)
```

Glavni program treba dopuniti sljedećim naredbama:

```
#kontrola: kružni isječak
gumb_isj = Button(prozor, text = 'Kružni isječak', command = kružni_isjecak)
gumb_isj.grid(column = 1, row = 3)
isj_poc = Label(prozor, text = 'Početni kut = ')
isj_poc.grid(column = 2, row = 3)
kutp = Entry(prozor, width = 7)
kutp.grid(column = 3, row = 3)
isj_zav = Label(prozor, text='Završni kut = ')
isj_zav.grid(column = 2, row = 4)
kutz = Entry(prozor, width = 7)
kutz.grid(column = 3, row = 4)
```

### 4. Kružni odsječak

Potprogram za crtanje kružnog odsječka:

```
def kružni_odsjecak():
    r1 = int(r.get())
    kut1 = int(kutp1.get())
    kut2 = int(kutz1.get())
    color('black', 'red')
    seth(kut1);
    fd(r1); lt(90)
    x = pos()
    begin_fill()
    circle(r1, kut2-kut1)
    setpos(x)
    end_fill()
    seth(90); pu(); setpos(0,0)
    lt(90); fd(r1); rt(90)
    pd()
```

Glavni program treba dopuniti sljedećim naredbama:

```
#kontrola: kružni odsječak
gumb_odsj = Button(prozor, text = 'Kružni odsječak', command = kružni_odsjecak)
gumb_odsj.grid(column = 1, row = 5)
odsj_poc = Label(prozor, text = 'Početni kut = ')
odsj_poc.grid(column=2, row = 5)
kutp1 = Entry(prozor, width = 7)
kutp1.grid(column = 3, row = 5)
odsj_zav = Label(prozor, text='Završni kut = ')
odsj_zav.grid(column = 2, row = 6)
kutz1 = Entry(prozor, width =7)
kutz1.grid(column = 3, row = 6)
```

## 5. Računanje opsega

Potprogram za računanje opsega:

```
def opseg():
    r1 = int(r.get())
    o = round(2*r1*pi,2)
    ispis_opseg.config(text = 'Opseg = ' + str(o))
```

Glavni program treba dopuniti sljedećim naredbama:

```
#kontrola: opseg
gumb_opseg = Button(prozor, text = 'Opseg', command = opseg)
gumb_opseg.grid(column = 1, row=7)
ispis_opseg = Label(prozor)
ispis_opseg.grid(column = 2, row = 7)
```

## 6. Računanje površine

Potprogram za računanje površine:

```
def površina():
    r1 = int(r.get())
    p = round(r1*r1*pi,2)
    ispis_površina.config(text='Površina = ' +str(p))
```

Glavni program treba dopuniti sljedećim naredbama:

```
#kontrola: površina
gumb_površina =Button(prozor, text='Površina', command=površina)
gumb_površina.grid(column=1, row=8)
ispis_površina = Label(prozor)
ispis_površina.grid(column=2, row=8)
```

 Rješenje u datoteci: **86Bvje20.py**.