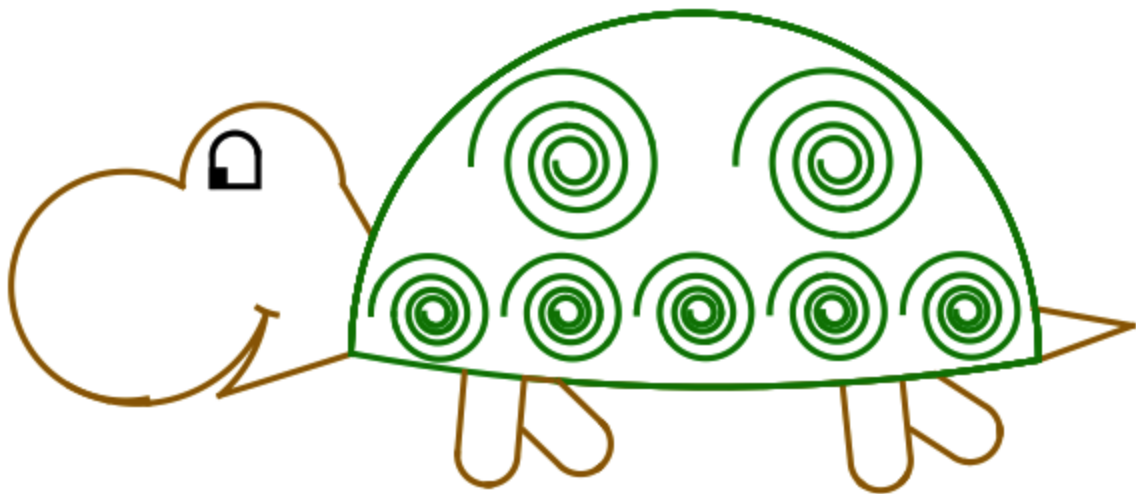


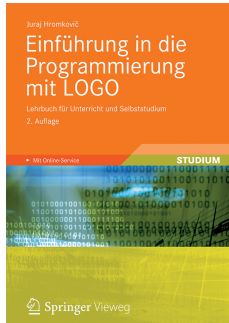
Heidi Gebauer Juraj Hromkovič Lucia Keller
Ivana Kosírová Giovanni Serafini Björn Steffen

Programiranje sa LOGO



Programiranje sa LOGO

Sadržaj ovog skripta se uzima iz udžbenika *Einführung in die Programmierung mit LOGO*, lekcije 1–7. Kompletan udžbenik se sastoji od 15 lekcija i sadrži mnogo dodatnih vežbi i objašnjenja, kao i vodič za učitelje.



Juraj Hromkovič. *Einführung in die Programmierung mit LOGO: Lehrbuch für Unterricht und Selbststudium*, 2. Aufl., Springer Vieweg 2012. ISBN: 978-3-8348-1852-2.

Verzija 3.0, 19. juli 2013, SVN-Rev: 12183

Prevod u srpski jezik: Marko Živković, Danijela Jovanović
Redakcija: Hans-Joachim Böckenhauer, Ivana Kosírová

Programsko Okruženje

Primeri i vežbe u ovom skriptu su namenjeni za XLogo programsko okruženje, koje se može besplatno skinuti sa internet stranice xlogo.tuxfamily.org.

Da bi primeri propisno radili, jezik u XLogo se mora podesiti na engleski.

Prava korišćenja

ABZ obezbeđuje predstavljen materijal besplatno za internu upotrebu i za obrazovne svrhe učitelja i obrazovnih institucija.

ABZ

Centar za nastavu informatike (ABZ) od ETH Cirih podržava škole i učitelje koji žele da uspostave i prošire obrazovne aktivnosti u oblasti informatike.

Podrške mogu da budu individualna konsultacija, nastava na licu mesta od profesora ETH-a i od tima ABZ-a, ili obuka za učitelje. Moguće je i obezbeđivanje nastavnog materijala.

www.abz.inf.ethz.ch

1 Osnovna Uputstva

Komanda je uputstvo koje kompjuter može da razume i da izvrši.

U principu, kompjuter jedino razume mnogo proste komande, koje mogu da se iskobinuju u oblik komplikovanijih uputstva. Ovakva sekvenca komandi se zove **kompjuterski program**.

Pisati kompjuterske programe nije jednostavno. Postoje programi koji se sastoje od više miliona komandi. Da bi zadržali pregled, mnogo je bitno da nastupimo zadatku pisanja komplikovanih programa strukturirano i dobro promišljeno. To ćemo da naučimo na ovom kursu.

Crtanje pravih linija

Komandom **forward 100** ili **fd 100** kornjača će ići 100 koraka unapred:



Sa komandom **back 100** ili **bk 100**, možeš da pomeriš kornjaču 100 koraka unazad:



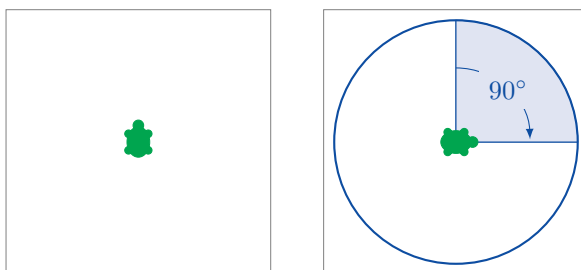
Brisanje i Restartovanje

Komanda **cs** briše čitav ekran i vraća kornjaču na početak.

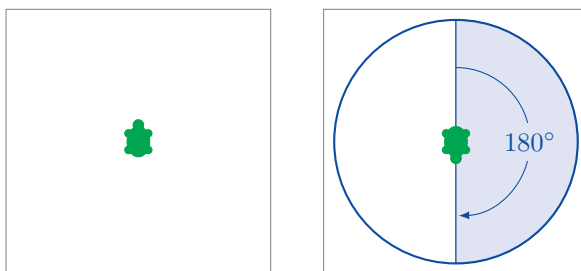
Skretanje

Kornjača se kreće uvek u pravcu prema kom je okrenuta.

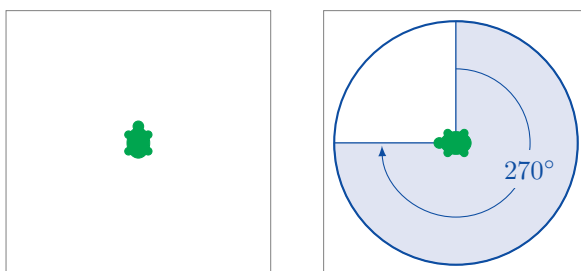
Koristeći komande **right 90** ili **rt 90**, možeš okrenuti kornjaču 90° na desno. To odgovara četvrtini kruga:



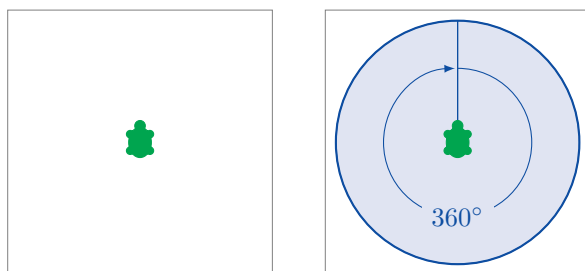
Komanda **right 180** ili **rt 180** okreće kornjaču 180° na desno. To odgovara polukružnom okretu:



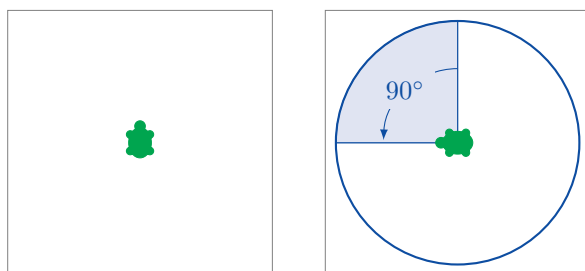
right 270 ili **rt 270** okreće kornjaču 270° na desno:



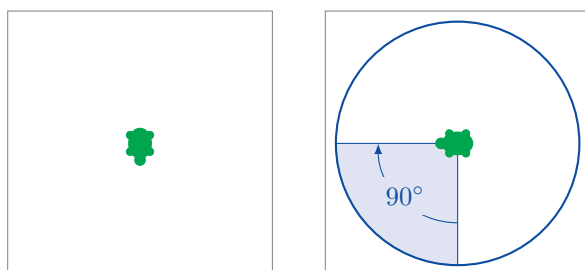
Komande **right 360** i **rt 360** okreću kornjaču 360° na desno. To odgovara punom okretu.



Koristeći komande **left 90** ili **lt 90**, kornjača će se okrenuti 90° na levo:



Napomenućemo da su pravci kretanja tumačeni sa tačke gledišta kornjače. To je ilustrirano u sledećem primeru koristeći komande **rt 90**:



Programiranje

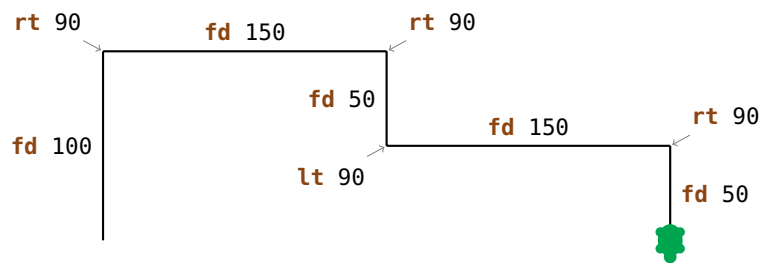
Programirati znači pisati više komandi u nizu.

Vežba 1

Ukucaj i izvrši sledeći program:

```
fd 100  
rt 90  
fd 150  
rt 90  
fd 50  
lt 90  
fd 150  
rt 90  
fd 50
```

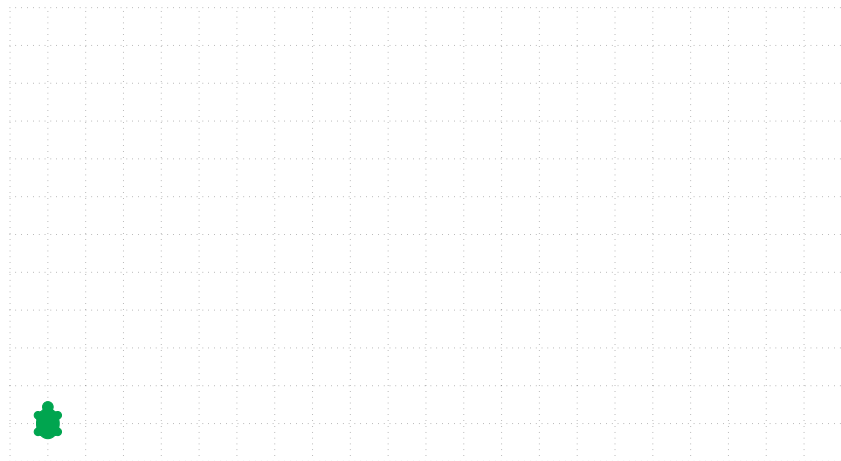
Da li si dobio sledeću sliku?



Vežba 2

Ukucaj i izvrši sledeći program:

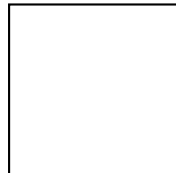
```
fd 100  
rt 90  
fd 200  
rt 90  
fd 80  
rt 90  
fd 100  
rt 90  
fd 50
```



Nacrtaj dobijenu sliku pored programa gore i opiši koja komanda je nacrtala koji deo slike (isto kao u vežbi 1).

2 Komanda **repeat**

Ako hoćemo da nacrtamo kvadrat sa stranicama dužine 100,



to možemo da uradimo sa sledećim programom:

```
fd 100  
rt 90  
fd 100  
rt 90  
fd 100  
rt 90  
fd 100  
rt 90
```

Napomenućemo da se sledeće dve komande

```
fd 100  
rt 90
```

ponavljaju četiri puta. Zar nebi bilo mnogo lakše kad bi kompjuteru rekli da ponovi te dve komande četiri puta, umesto da ih pišemo četiri puta jednu za drugom?

Baš to možemo da uradimo ovako:

repeat	4	[fd 100 rt 90]
Komanda za ponavljanje jednog programa	Broj za ponavljanje	Sekvenca komande za ponavljanje

Vežba 6

Ukucaj i izvrši sledeći program:

```
fd 75 lt 90  
fd 75 lt 90  
fd 75 lt 90  
fd 75 lt 90
```

Koju sliku će nacrtati ovaj program? Da li možeš da upotrebiš komandu **repeat** da skратиš ovaj program?

Vežba 7

Ukucaj sledeći program i saznaj šta će izazvati.

```
fd 50 rt 60  
fd 50 rt 60  
fd 50 rt 60  
fd 50 rt 60  
fd 50 rt 60  
fd 50 rt 60
```

Skrati ovaj program koristeći komandu **repeat**.

Vežba 8

Koristi komandu **repeat** u programu koji će nacrtati kvadrat sa stranicama dužine 200.

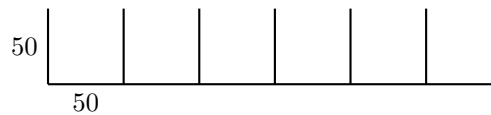
Vežba 9

Ukucaj sledeći program:

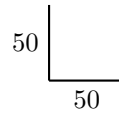
```
fd 100 rt 120  
fd 100 rt 120  
fd 100 rt 120
```

Koji je rezultat kad izvršavas ovaj program? Koristi komandu **repeat** da skратиš ovaj program.

Hoćemo da nacrtamo sledeću sliku koristeći komandu **repeat**:



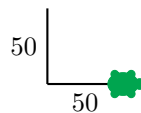
Pre nego što počnemo da crtamo, moramo da razmislimo koji šablon se ponavlja. Na primer, mogli bi da koristimo sledeću sliku kao šablon za ponavljanje:



Ako počnemo od ugla dole levo, ova slika se može nacrtati sledećim programom.

```
fd 50 bk 50 rt 90 fd 50
```

Posle izvršavanja programa, kornjača će se nalaziti na mestu kao u sledećoj slici pokazano, i gledaće na desno.

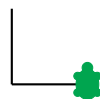


Sve što sad još moramo da uradimo, to je da okrenemo kornjaču na gore, tako da možemo da nastavimo sa crtanjem sledećeg šablona. Da bi postigli to, možemo koristiti komandu **lt 90**.

Izvršićemo celi program da bi proverili šta smo postigli do sad.

```
fd 50 bk 50 rt 90 fd 50  
lt 90
```

Dobićemo željeni rezultat.



Ako sad izvršimo taj isti program još jednom, dobićemo sledeće:



Sad možemo da vidimo da naša ideja radi i da možemo ponavljati ovaj program 6 puta:

```
repeat 6 [ fd 50 bk 50 rt 90 fd 50 lt 90 ]
```

šablon
orijentisanje

Mnogo vežbe se mogu rešiti ovim pristupom.

Uvek zapamti da prvo trebaš naći šablon za ponavljanje. Onda napiši jedan program za crtanje *šablona* i drugi program za *orijentisanje* kornjače tako da gleda u tačan pravac za sledeće ponavljanje šablona.

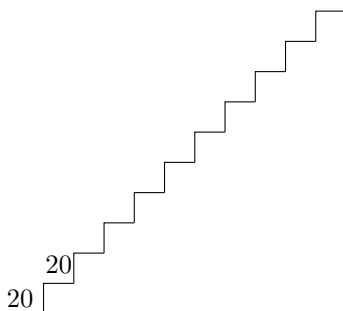
Tvoj program će onda izgledati ovako:

repeat *Broj ponavljanja* [*šablon orijentisanje*]

Vežba 10

Crtanje stepeništa.

(a) Nacrtaj stepenište od 10 stepenika veličine 20.



- Prvo najdi ponavljajući šablon i napiši program koji će ga nacrtati.
- Razmisli kako napisati program koji će okrenuti kornjaču u tačan pravac za sledeće ponavljanje šablona.
- Sastavi oba programa da rešiš zadatak.

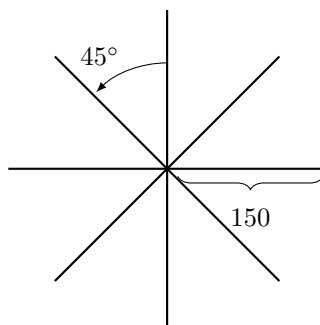
(b) Nacrtaj stepenište od 5 stepenika veličine 50.

(c) Nacrtaj stepenište od 20 stepenika veličine 10.

Vežba 11

Sada ćemo da crtamo zvezde.

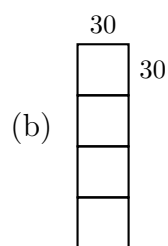
(a) Nacrtaj sledeću zvezdu.



(b) Ova zvezda ima osam kraka veličine 150. Možeš li da nacrtáš zvezdu sa 16 kraka veličine 100?

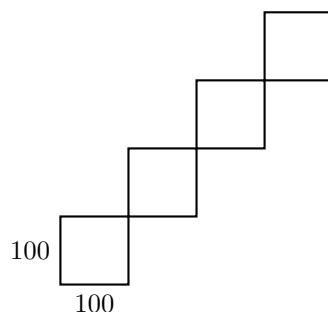
Vežba 12

Nacrtaj sledeće slike programom:



Vežba 13

Nacrtaj sledeću sliku programom:



Vežba 14

Ukucaj i izvrši sledeći program:

```
repeat 4 [fd 100 rt 90]
rt 90
repeat 4 [fd 100 rt 90]
rt 90
repeat 4 [fd 100 rt 90]
rt 90
repeat 4 [fd 100 rt 90]
rt 90
```

Koji je rezultat? Možeš li napisati ovaj program još kraće?

Režim Hodanja

Naša kornjača se obično nalazi u **režimu crtanja**. To znači da kornjača ima olovku vezanu za sebe i kad god se kreće, povlači liniju.

Međutim, u **režimu hodanja** kornjača se kreće bez crtanja. Možeš prebaciti u režim hodanja koristeći sledeću komandu:

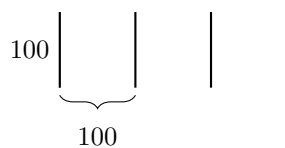
penup ili **pu** ukratko.

Da prebacimo nazad u režim crtanja, koristimo sledeću komandu:

pendown ili **pd** ukratko.

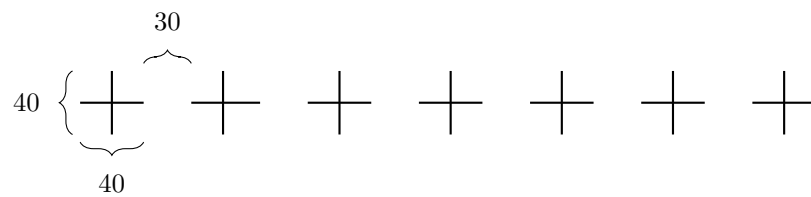
Vežba 15

Nacrtaj sledeću sliku programom:



Vežba 16

Napiši program koji će nacrtati sledeću sliku:



3 Imenovanje i Pozivanje Programa

Svakom programu, koji smo do sad napisali, možemo dati ime. Ako nakon toga napišemo to ime u komandnu liniju, program će biti izvršen.

Program za crtanje kvadrata sa stranicama veličine 100 je:

```
repeat 4 [fd 100 rt 90]
```

Možemo imenovati ovaj program **KVADRAT100** ovako:

```
to KVADRAT100
repeat 4 [fd 100 rt 90]
end
```

Napisali smo isti program dvaput. Jednom sa i jednom bez imenovanja.

Da bi napisali program sa imenom, moramo koristiti **editor**. Takvi programi će biti označeni sivim poljem.

čim završimo pisanje programa, moramo pritisnuti dugme sa kornjačom da bi zatvorili editor.

Svako može sam da izabere ime za program. Mi smo izabrali **KVADRAT100**, jer smo hteli da ukažemo da program crta kvadrat sa stranicama veličine 100.

Ograničenje je da to ime sme samo da se sastoji od slova i cifri, i da je samo jedna jedina reč (ne smeš koristiti razmak).

Posle pisanja programa, još ništa neće biti nacrtano. Mi smo tek imenovali program, ali nismo ga još izvršili. Ako sad ukucamo njegovo ime

KVADRAT100

u komandnu liniju, program **repeat 4 [fd 100 rt 90]** će biti izvršen. Ekran će pokazati:



Vratimo se na vežbu 12(a). Mogli bi da pojednostavimo rešenje ako bi prvo napisali program za ponavljajući šablon, kvadrat sa stranicama veličine 20, i onda dati mu prikladan naziv:

```
to KVADRAT20
repeat 4 [fd 20 rt 90]
end
```

Posle crtanja **KVADRAT20**, kornjača će se nalaziti na donjem levom uglu kvadrata:



Da bi nacrtali sledeći kvadrat, moramo pomeriti kornjaču do donjeg desnog ugla. To možemo učiniti sledećim programom:

```
rt 90 fd 20 lt 90
```

Označimo i ovaj program:

```
to POMERI20
rt 90 fd 20 lt 90
end
```

Koristeći ta dva programa, možemo napisati program za vežbu 12(a) ovako:

```
repeat 10 [KVADRAT20 POMERI20]
```

Ovaj program možemo isto imenovati. Na primer:

```
to RED10
repeat 10 [KVADRAT20 POMERI20]
end
```

Programi **KVADRAT20** and **POMERI20** se zovu **potprogrami** programa **RED10**.

Vežba 17

Napiši program koji rešava vežbu 12(b) i koji koristi jedan potprogram koji crta kvadrat sa stranicama veličine 30. Tvoj završni program treba da izgleda ovako:

```
repeat 4 [KVADRAT30 POMERI30]
```

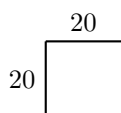
Dakle, moraš da napišeš potprograme **KVADRAT30** and **POMERI30**.

Vežba 18

Koristi program **KVADRAT100** kao potprogram da nacrtáš sliku od vežbe 13.

Vežba 19

Napiši program koji crta stepenik



i koristi ga kao potprogram za rešavanje vežbe 10(a).

Vežba 20

Nađi drugo rešenje za vežbu 11(a) koristeći sledeći program kao potprogram:

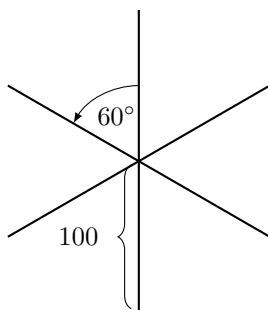
```
to LINIJA  
fd 150 bk 150  
end
```

Vežba 21

Napiši sledeći program u editoru:

```
to KRAK  
fd 100 bk 200 fd 100  
end
```

Koristi program **KRAK** kao potprogram programa **ZVEZDA6** da nacrtáš sledeću sliku:



Vežba 22

Reši vežbu 15 i vežbu 16 ponovo koristeći potprograme.

Vežba 23

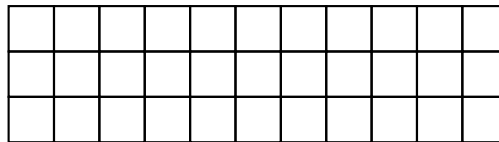
Već smo napisali program **RED10**. šta radi sledeći program?

```
RED10 fd 20 lt 90 fd 200 rt 90
```

Testiraj tvoju ideju koristeći kompjuter.

Vežba 24

Napiši program koji crta sledeću sliku:



Vežba 25

Nacrtaj kvadrate različitih veličina.

- (a) Napiši program koji crta kvadrate sa stranicama veličine 50 i imenuj ga **KVADRAT50**. Izvrši program i proveriti da li radi kao što je očekivano.
- (b) Napiši program koji crta kvadrat sa stranicama veličine 75.
- (c) Izvrši program

```
KVADRAT50  
KVADRAT75  
KVADRAT100
```

Kako izgleda dobijena slika?

- (d) Kako možeš da promeniš gore navedeni program da bi nacrtao tri još veća kvadrata?

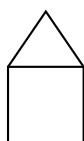
Gradnja Kuće

U sledećem, pomoći ćemo arhitekti da sagradi stambeni kompleks. Da bi gradnja bila što lakša, on planira da gradi sve kuće iste.

Predlažemo sledeći dizajn:

```
to KUCA
rt 90
repeat 4 [fd 50 rt 90]
lt 60 fd 50 rt 120 fd 50 lt 150
end
```

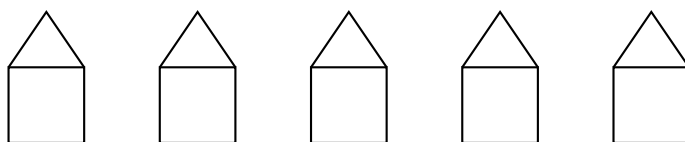
Ovaj program crta sledeću kuću:



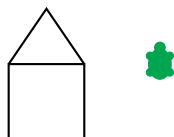
Vežba 26

Gde počinje kornjača sa crtanjem kuće? Razmisli kojom će putanjom kornjača ići dok crta kuću koristeći program **KUCA**. Gde će se kornjača nalaziti na kraju izvršavanja? Nacrtaj sliku i opiši efekat svake komande isto kao što smo radili u vežbi 1.

Arhitekta gradi kuće našim programom i zadovoljan je rezultatom. Zato on želi da koristi taj program da sagradi ceo blok sa kućama. Završni blok treba ovako izgledati:



Pošto sve kuće izgledaju isto, on može da koristi isti program **KUCA** 5 puta bez razmišljanja o planu za svaku kuću. On će pustiti kornjaču da počne sa kućom skroz levo, pa će joj reći da se pomeri do početka druge kuće:



Arhitekta to radi sledećim programom:

```
KUCA rt 90 pu fd 50 lt 90 pd
```

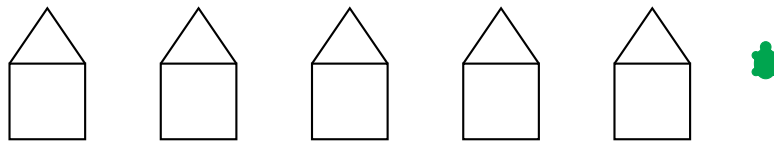
Pošto se kornjača nalazi na tačnom mestu, ona može da nacрта drugu kuću isto kao što je nacrtala prvu i pomeriti se do početka sledeće kuće.

Taj proces se treba ponavljati sve dok sve pet kuće ne budu nacrtane. Znači trebamo ponavljati gore navedeni program pet puta da bi dobili red sa pet kuća.

Završni program će se zvati **REDKUCA**:

```
to REDKUCA
repeat 5 [KUCA rt 90 pu fd 50 lt 90 pd]
end
```

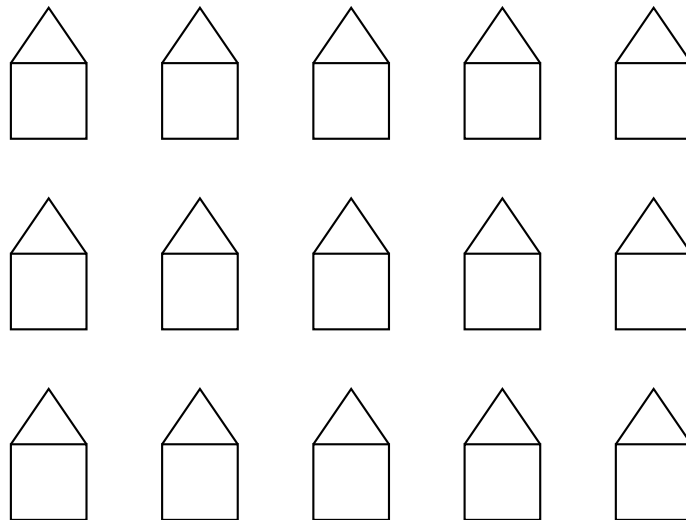
Posle izvršenja, kornjača se nalazi na tački gde bi sledeća kuća bila nacrtana:



Vežba 27

U ovom trenutku želimo da proširimo naš stambeni kompleks dodajući više ulica.

Koristi program **REDKUCA** kao kamen temeljac za crtanje sledeće slike:



Nagoveštaj: Nakon završetka reda, kornjača mora da se pomeri na tačnu poziciju za početak crtanja sledećeg reda.

Debele Linije i Crni Kvadrati

Vežba 28

Koristiti **DEBELA** za crtanje debelih linija.
Imenuj sledeći program imenom **DEBELA**

```
fd 100  
rt 90  
fd 1  
rt 90  
fd 100  
rt 180
```

i napiši u komandnoj liniji

DEBELA

šta će kornjača nacrtati? Nacrtaj olovkom kako je ta slika formirana.

Vežba 29

Ponovi program **DEBELA** 100 puta koristeći sledeće komande

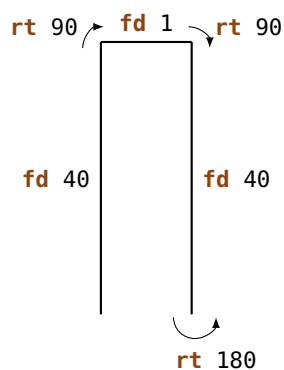
```
repeat 100 [DEBELA]
```

Kako izgleda dobijena slika?

Vežba 30

U ovoj vežbi ćemo crtati debele linije. U vežbi 28, već smo videli da debela linije može biti nacrtana ovako:

```
to DEBELA40  
fd 40  
rt 90  
fd 1  
rt 90  
fd 40  
rt 180  
end
```



Dve obične linije su nacrtane blizu zajedno da bi izgledale kao jedna debela linija.

Ukucaj program **DEBELA40** i probaj ga.

Vežba 31

Debela linija dužine 40 može se gledati kao pravougaonik širine 1 i dužine 40. Posle izvršenja **DEBELA40** kornjača gleda na gore. Izvršavanjem programa još jednom, druga linija će biti precrtana. Dobićemo pravougaonik širine 2 i dužine 40. Svako dodatno izvršenje dodaje jednu liniju.

Ponavljajući **DEBELA40** 40 puta, dobijamo ispunjen kvadrat sa stranicama veličine 40. Probaj to ponavljajući **DEBELA40** 40 puta.

Napiši program sa imenom **CRN040**, koji crta crni kvadrat sa stranicama veličine 40.

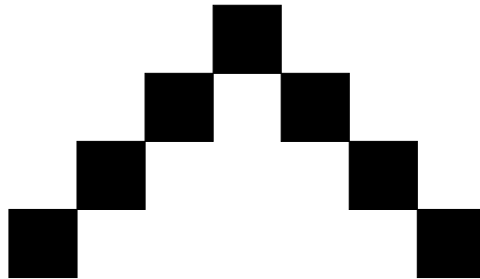
Vežba 32

Nacrtaj sledeću sliku koristeći program **CRN040**:



Vežba 33

Koristi program **CRN040** da nacrtáš sledeću sliku:



Vežba 34

Nacrtaj sledeću sliku:



Vežba 35

Napiši program koji crta sledeću sliku:



Vežba 36

Arhitekta je odlučio da naruči krov za kuću kod drugog prodavca. Znači dobiće dva tipa kamena temeljca: Jedan se zove **KROV** a drugi se zove **OSNOVA**.

Napiši dva programa koji će nacrtati ta dva kamena temeljca. Kombinuj ta dva programa u jednom novom programu **KUCA1** koji crta kuću.

Vežba 37

Kuće u vežbi 27 su skroz jednostavne. Budi kreativan i smisli novi dizajn za kuću. Koristi svoju kuću da sagradiš celi stambeni kompleks.

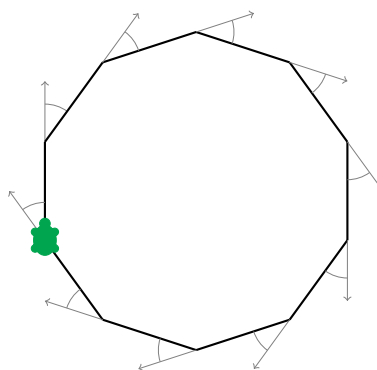
4 Pravilni Mnogouglovi i Krugovi

Pravilni Mnogouglovi

Pravilni k -ugao je mnogougao sa k uglovima i k stranicama jednake veličine. Da bi nacrtao pravilni 10-ugao olovkom, moraš nacrtati 10 linije i posle svake linije moraš da promeniš pravac “pomalo”.

Koliko moramo da se okrenemo?

Kada crtamo jedan pravilan mnogougao, okrećemo kornjaču više puta, ali na kraju ona uvek stiže na početnu tačku i gleda u početni smer.



To znači da se kornjača okrenula ukupno punih 360° dok je crtala mnogougao. Kada crtamo pravilan 10-ugaonik, mi se okrećemo tačno deset puta, i svaki put pod istim uglom.

Ugao pod kojim se okrećemo je zato:

$$\frac{360^\circ}{10} = 36^\circ$$

Prema tome moramo okrenuti kornjaču za 36° svaki put: **rt 36**. To ćemo probati sledećim programom:

```
repeat 10 [ fd 50      rt 36 ]  
           dužina stranice  okrenuti za 36°
```


Vežba 38

Nacrtaj sledeće pravilne mnogouglove:

- (a) pravilan 5-ugao sa stranicama dužine 180,
- (b) pravilan 12-ugao sa stranicama dužine 50,
- (c) pravilan 4-ugao sa stranicama dužine 200,
- (d) pravilan 6-ugao sa stranicama dužine 100,
- (e) pravilan 3-ugao sa stranicama dužine 200,
- (f) pravilan 18-ugao sa stranicama dužine 20.

Kada crtamo pravilan 7-ugao, nailazimo na problem da 360 nije deljiv sa 7 bez ostatka. U tim slučajevima, neka kompjuter izračuna rezultat za nas, pisanjem

`360/7`

("/" znači za kompjuter da "deli"). Kompjuter će onda izračunati rezultat. Dakle, možemo crtati pravilan 7-ugao sa stranicama dužine 100 ovako:

```
repeat 7 [fd 100 rt 360/7]
```

Isprobaj to.

Crtanje Krugova

Ne možemo crtati precizne krugove koristeći samo **fd** i **rt**. Međutim, možda ste primetili, da pravilan mnogougao sa mnogo uglova izgleda skoro kao jedan krug. Znači možemo crtati krugove mnogouglovima sa mnogo kratkim stranicama i puno uglova.

Vežba 39

Isprobaj sledeće programe:

```
repeat 360 [fd 1 rt 1]  
repeat 180 [fd 3 rt 2]  
repeat 360 [fd 2 rt 1]  
repeat 360 [fd 3.5 rt 1]
```

3.5 znači 3 i po koraka.

Vežba 40

- (a) Kako bi nacrtao mnogo mali krug? Napiši program za to.
- (b) Kako bi nacrtao veliki krug? Napiši program za to.

Vežba 41

Pokušaj da nacrtáš sledeće polukrugove. Možeš sam izabrati veličine:



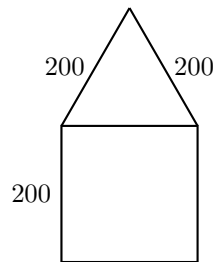
(a)



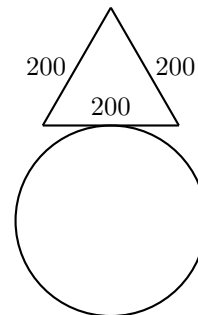
(b)

Vežba 42

Koristi to što si naučio, da bi nacrtao sledeće slike. Možeš sam izabrati veličine krugova:



(a)



(b)

Slobodno Crtanje

Nacrtaj 7-ugao:

```
repeat 7 [fd 100 rt 360/7]
```

Onda okreni kornjaču za 10° koristeći komandu

```
rt 10
```

Ponavljaj oba programa nekoliko puta i posmatraj dobijenu sliku. Posle svakog 7-ugla okrećemo kornjaču za 10° sa **rt 10**. Ako želimo da se vratimo na početnu tačku, moramo to da ponavljamo

$$\frac{360^\circ}{10^\circ} = 36$$

puta. Zato želimo da vidimo šta sledeći program radi:

```
repeat 36 [repeat 7 [fd 100 rt 360/7] rt 10]
```

Vežba 43

Nacrtaj pravilan mnogougao sa 12 ugla i stranicama dužine 70. Okreni ga 18 puta dok se ne vrati na početnu tačku.

Napomena: Možeš prvo napisati program za crtanje 12-ugla sa stranicama dužine 70 i imenovati ga **UGA012**, na primer. Posle toga samo moraš da završiš sledeći program:


















```
repeat 18 [UGA012 rt ... ]
```

Vežba 44

Izmisli vežbu sličnoj vežbi 43. Napiši program koji rešava tvoju vežbu.

Boje

Da bi nacrtali lepe slike, trebaju nam različite boje. Kornjača ne zna samo da crta crne linije, nego i linije mnogo različitih boja. Svakoj boji je dodeljen jedan broj. Sledeća tabela sadrži listu svih mogućih boja:

0		5		9		13	
1		6		10		14	
2		7		11		15	
3		8		12		16	
4							

Koristeći komandu

setpencolor	X
Komanda za menjanje boje	Broj za že- ljenu boju

kornjača menja sadašnju boju u boju sa brojem **X**. Možemo koristiti kraću verziju te komande: **setpc**.

Koristeći boje, možemo nacrtati neverovatne šare kao na primer sledećim programom. Prvo napravimo dva imenovana programa koji crtaju krugove različitih veličina.

```
to KRUG3
repeat 360 [fd 3 rt 1]
end

to KRUG1
repeat 360 [fd 1 rt 1]
end
```

Onda ćemo koristiti te krugove da dizajniramo šare slične onima koje smo već videli:

```
to SARA3
repeat 36 [KRUG3 rt 10]
end

to SARA1
repeat 18 [KRUG1 rt 20]
end
```

Hajde da pokušamo istu stvar pomoću boja:

```
setpc 2
SARA3 rt 2
setpc 3
SARA3 rt 2

setpc 4
SARA3 rt 2
setpc 5
SARA3 rt 2
```

setpc 6
SARA1 **rt** 2
setpc 15
SARA1 **rt** 2

setpc 8
SARA1 **rt** 2
setpc 9
SARA1 **rt** 2

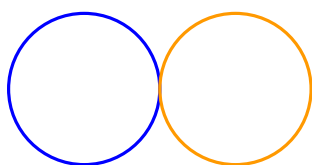
Smeš slobodno da nastaviš i da dodaš još više šare. Možeš i izmisliti skroz nove šare.

Vežba 45

Koristi program **SARA3** da nacrtáš odgovarajuću šaru narandžastom bojom. Onda koristi komandu **setpc** 7 da izabereš belu boju. Šta će se desiti ako izvršiš **SARA3** još jednom?

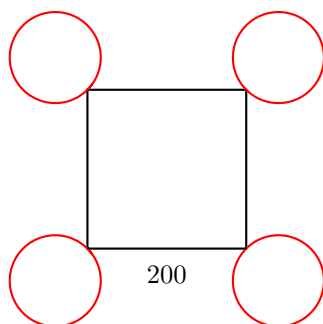
Vežba 46

Nacrtaj sledeću sliku. Kornjača počinje na zajedničkoj tački oba kruga (njihova tačka preseka).



Vežba 47

Napiši program koji crta sledeću sliku. Možeš sam izabrati veličinu krugova.



5 Programi sa Parametrima

U lekciji 3, naučili smo kako se mogu imenovati programi i kako se ta imena mogu koristiti za pozivanje tih programa i crtanje željenih slika.

Onda smo naučili u lekciji 4 kako se mogu crtati pravilni mnogouglovi. Pisati programe za svaki mnogougao je monotono jer se programi samo razlikuju u broju uglova.

Hajmo da pogledamo sledeća tri programa:

```
repeat 7 [fd 50 rt 360/7]
repeat 12 [fd 50 rt 360/12]
repeat 18 [fd 50 rt 360/18]
```

Sva tri programa su mnogo slična i samo se razlikuju po žutim brojevima 7, 12 i 18. Ti brojevi definišu broj uglova mnogougla.

U sledećem, hoćemo da napišemo program koji radi za bilo koji mnogougao, bez obzira koliko uglova on ima:

```
to MNOGOUGAO :UGLA
repeat :UGLA [fd 50 rt 360/:UGLA]
end
```

šta smo uradili? Kad god se broj uglova pojavi u programu za mnogougao, mi smo napisali ime :UGLA umesto stvarnog broja. Da bi kompjuter znao da mi želimo da biramo broj uglova slobodno, moramo napisati :UGLA i posle imena programa.

Kada kucamo MNOGOUGAO 12 u komandnu liniju, kompjuter će izmeniti ime :UGLA sa brojem 12 gde god se to ime pojavi:

```
repeat 12 :UGLA [fd 50 rt 360/12]
```

Isprobaj to:

```
MNOGOUGAO 3
MNOGOUGAO 4
MNOGOUGAO 5
MNOGOUGAO 6
```

:UGLA se zove **parametar**. U primeru gore, vrednosti 3, 4, 5 i 6 se zovu **vrednosti parametra :UGLA**. Kompjuter zna da je ovo parametar zbog **:**. Zato, svaki put kad se pojavi jedan parametar, on mora da ima **:** ispred imena.

Vežba 48

Svaki od sledećih programa crta kvadrat različitih veličina.

```
repeat 4 [fd 100 rt 90]
repeat 4 [fd 50 rt 90]
repeat 4 [fd 200 rt 90]
```

Žuti brojevi 100, 50, 200 se mogu gledati kao parametri koji podešavaju veličinu kvadrata. Napiši program sa jednim parametrom **:VELICINA** za crtanje kvadrata bilo koje veličine:

```
to KVADRAT :VELICINA
...
end
```

Vežba 49

Sledeći programi crtaju krugove različitih veličina:

```
repeat 360 [fd 1 rt 1]
repeat 360 [fd 12 rt 1]
repeat 360 [fd 3 rt 1]
```

Napiši program sa jednim parametrom da nacrtáš krugove bilo koje veličine. Isprobaj tvoj program koristeći 1, 2, 3, 4 i 5 kao različitu vrednost parametra. Možeš sam da biraš ime za parametar, ali zapamti da ispred parametra mora imati **:**.

Vežba 50

Da li se još sećaš kako da nacrtáš debele linije (vežba 28)? Napiši program sa jednim parametrom koji može da crta debele linije bilo koje veličine. *Nagoveštaj*: Možeš početi sa pisanjem programa koji crtaju linije dužine 100 i dužine 50 da shvatiš šta parametar tvog programa treba da bude.

Vežba 51

Napiši program sa jednim parametrom koji crta pravilne trouglove bilo koje veličine. Koristi tvoj program da nacrtáš trouglove veličine

20, 40, 60, 80, 100, 120, 140, 160 i 180

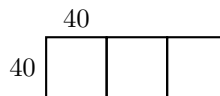
jedan za drugim. šta dobijaš?

Vežba 52

Sada želimo da nacrtamo kvadrate veličine 40, jedan do drugog. Napiši program **KVADRATI** sa jednim parametrom **:BROJ**. Parametar **:BROJ** određuje koliko kvadrata treba da se nacrtaju. Dakle, kada pozovemo **KVADRATI 6**, kornjača će nacrtati sledeću sliku:

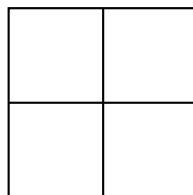


Ova slika će biti nacrtana posle pozivanja **KVADRATI 3**:



Vežba 53

Napiši program koji će nacrtati sliku koja se sastoji od 4 kvadrata. Jedan parametar treba da odredi veličinu kvadrata.

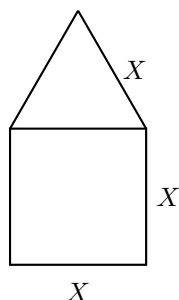


Vežba 54

Napiši program koji će nacrtati pravilan 6-ugao bilo koje veličine. Isprobaj tvoj program crtanjem pravilnog 6-ugla veličina 40, 60 i 80.

Vežba 55

Napiši program sa parametrom `:X` koji će nacrtati kuće sledećeg tipa:



Programi sa Više Parametra

Programi mogu imati više od jednog parametra. Kada crtamo mnogouglove, na primer, naš program bi mogao imati jedan parametar `:UGLA` za broj uglova, a drugi parametar `:VELICINA` za dužinu stranica.

U sledećim programima, parametar `:UGLA` je obeležen žutom bojom a parametar `:VELICINA` je obeležen zelenom bojom:

```
repeat 13 [fd 100 rt 360/13]
repeat 3 [fd 300 rt 360/3]
repeat 17 [fd 10 rt 360/17]
repeat 60 [fd 3 rt 360/60]
```

Onda možemo napisati program sa dva parametra koji može crtati pravilne mnogouglove:

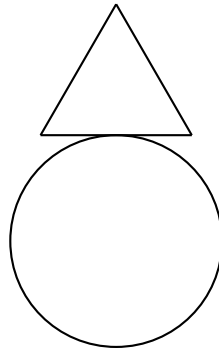
```
to MNOGOUGAO :UGLA :VELICINA
repeat :UGLA [fd :VELICINA rt 360/:UGLA]
end
```

Testiraj program `MNOGOUGAO` koristeći sledeće pozive:

```
MNOGOUGAO 12 60
MNOGOUGAO 12 45
MNOGOUGAO 8 30
MNOGOUGAO 9 30
MNOGOUGAO 7 31
MNOGOUGAO 11 50
```

Vežba 56

Napiši program sa dva parametra koji može nacrtati sledeću sliku. Veličinu kruga kao i veličinu trougla možeš sam izabrati.



Vežba 57

Program

```
fd 100 rt 90 fd 200 rt 90 fd 100 rt 90 fd 200
```

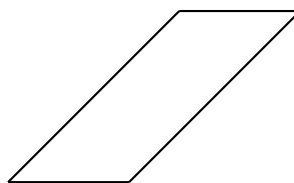
crta pravougaonik visine 100 i širine 200. Ukucaj ovaj program da bi video da li stvarno radi. Napiši drugi program sa dva parametra koji može nacrtati pravougaonike bilo koje visine i širine.

Vežba 58

Sledeći program

```
repeat 2 [rt 45 fd 200 rt 45 fd 100 rt 90]
```

crta paralelogram:



Napiši program sa dva parametra koji može da nacrtava ovakav paralelogram sa stranicama bilo koje veličine.

Vežba 59

Nacrtaj cveće ovako. Počni sa jednim krugom koristeći

```
MNOGOUGAO 360 2
```

onda okreni kornjaču samo malo na desno

```
rt 20
```

i nacrtaj drugi krug

```
MNOGOUGAO 360 2
```

Ponovi to više puta:

```
rt 20 MNOGOUGAO 360 2 rt 20 MNOGOUGAO 360 2 ...
```

Kada je cveće završeno, kornjača bi trebala biti na početnoj tački. Kornjača treba da je napravila 18 krugova sa okretanjem od 20° između svakog od njih. Ukupno se kornjača okrenula za $18 \cdot 20^\circ = 360^\circ$.

Možemo da napišemo ceo program kao:

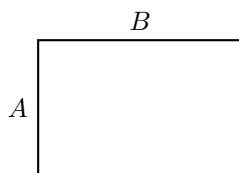
```
repeat 18 [MNOGOUGAO 360 2 rt 20]
```

Pokušaj to.

- Takođe možeš da nacrtáš cvet sa 10 ili čak 20 latica (krugova). Kako to uraditi? Napiši program i isprobaj ga.
- Možeš li napisati program sa jednim parametrom koji crta cvet sa bilo kojim brojem latica (krugova)?
- Da li možeš da napišeš program koji koristi vrednosti parametra za:
 - broj latica (krugova) i
 - veličinu krugova?

Vežba 60

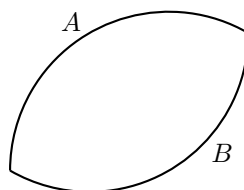
Napiši program da nacrtáš bilo koji pravougaonik bilo koje boje:



To znači da ne samo visina A i širina B , nego i boja mora da može slobodno da se izabere.

6 Crtati Cveće i Proslediti* Parametre Potprogramima

Jedan list



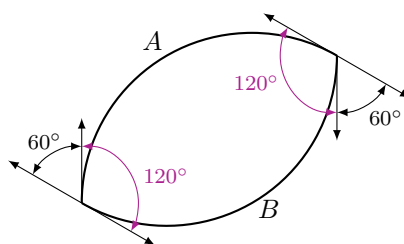
se može gledati kao dva luka A i B koji su zalepljeni jedan za drugog. Možemo koristiti sledeći program za crtanje jednog luka:

```
repeat 120 [fd 2 rt 1]
```

Isprobaj to.

Primećujemo da je program mnogo sličan programu za krugove. Umesto uraditi 360 malih koraka sa malim uglovima između, možemo ponavljati 120 puta [**fd** 2 **rt** 1] i tako crtati samo trećinu jednog kruga ($\frac{360}{3} = 120^\circ$).

Ostaje nam pitanje, koliko treba da okrenemo kornjaču pre nego što možemo početi sa crtanjem drugog luka B , koji će formirati donji deo našeg lista. Pogledajmo sledeću ilustraciju:



Ako želimo da se vratimo na početnu tačku posle crtanja celog lista, moraćemo da okrećemo kornjaču ukupno za 360° .

Dok crtamo deo A , okrećemo kornjaču za 120° i dok crtamo deo B , okrećemo je još jednom za 120° . Zato ostatak iznosi

$$360^\circ - 120^\circ - 120^\circ = 120^\circ.$$

Zato delimo ostatak od 120° jednako na dve rotacije kod šiljaka lista:

$$\frac{120^\circ}{2} = 60^\circ.$$

Napokon dobijamo sledeći program:

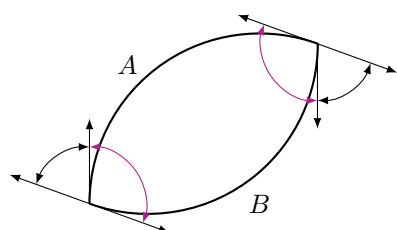
```
repeat 120 [fd 2 rt 1]
rt 60
repeat 120 [fd 2 rt 1]
rt 60
```

ili još jednostavnije:

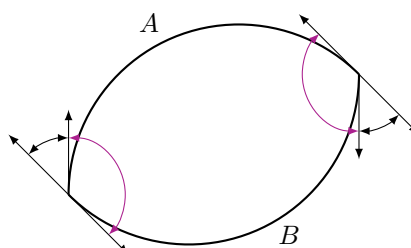
```
repeat 2 [repeat 120 [fd 2 rt 1] rt 60]
```

Isprobaj to.

Sada bi želili da crtamo uže lišće (gde su delovi A i B kraći) ili šire lišće (gde su delovi A i B duži).



UGA0=120°



UGA0=135°

Korisićemo opet program sa parametrom za ovo. Nazvaćemo parametar **:UGA0**. Taj ugao za okretati kornjaču na šiljku lista se može izračunati ovako:

Pre nego što počnemo sa delom B od lista, moramo završiti polovinu ukupne rotacije, to jeste $\frac{360^\circ}{2} = 180^\circ$. Ugao kojim se moramo okrenuti na šiljku lista je dat sa

$$180^\circ - \text{:UGA0}.$$

Sad Možemo napisati naš program u editoru:

```
to LISTIC :UGA0
repeat 2 [repeat :UGA0 [fd 2 rt 1] rt 180-:UGA0]
end
```

(Koristimo ime **LISTIC**, zato što komanda **list** već postoji u Logo jeziku.)

Isprobaj program u komandnoj liniji ovako:

```
LISTIC 20  
LISTIC 40  
LISTIC 60  
LISTIC 80  
LISTIC 100
```

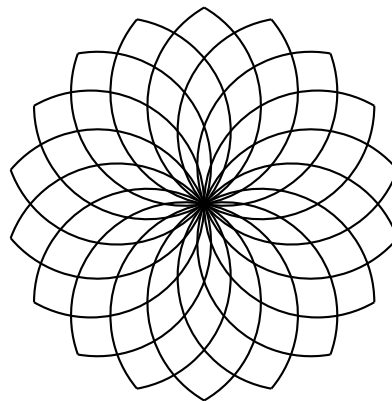
Šta se dešava?

Vežba 61

Slobodno crtanje.

Počni sa crtanjem jednog cveta koristeći sledeći program:

```
LISTIC 100  
rt 20  
LISTIC 100  
rt 20  
LISTIC 100  
....
```



Koliko puta moraš da ponavljaš komande **LISTIC** i **rt 20** za crtanje jednog čitavog cveta?

Napiši program za cvet u samo jednoj liniji koristeći prikladnu **repeat** komandu. (Nemoj da zaboraviš da sve rotacije **rt** između dva listića moraju ukupno da iznose 360°).

Vežba 62

Komanda **fd** 2 u programu **LISTIC** definiše veličinu kruga, iz kojeg smo isekli jedan luk. Možemo izmeniti vrednost 2 sa parametrom sa imenom **:VELICINA**. Napiši jedan program

```
LISCE :UGAO :VELICINA
```

sa dva parametra **:UGAO** i **:VELICINA**, tako da možemo podesiti dužinu luka i veličinu lišća. Isprobaj tvoj program sledećim pozivima:

```
LISCE 100 1
```

```
LISCE 100 1.5
```

```
rt 100
```

```
LISCE 80 2
```

```
LISCE 80 2.5
```

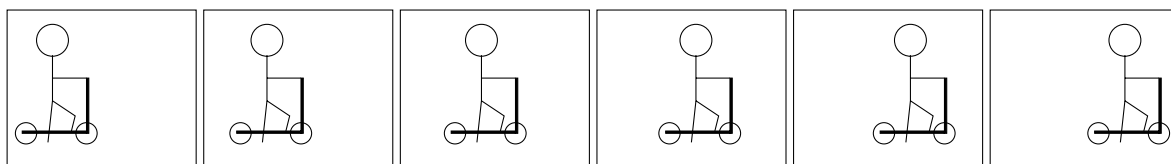
Onda okreni kornjaču za 80° na desno i ponovi gornje komande.

Vežba 63

Izmisli druge šare i nacrtaj ih.

7 Programirati Animacije

Znaš li kako možeš napraviti crtani film? To funkcioniše isto kao jedan kineograf. Prvo crtamo nekoliko slike koje se razlikuju samo pomalo jedna od druge. U sledećem primeru, jedan dečko na trotinetu se pomera samo pomalo od slike do slike:

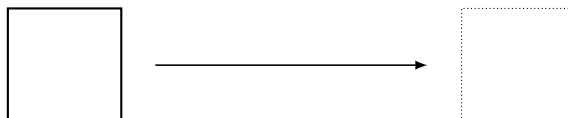


Kada staviš te slike jednu na drugu i kada palcem brzo prelistaš preko njih, onda ti se čini kao da se taj dečko stvarno pomera s leva na desno. Pokretne slike se zovu **animacije**.

U ovoj lekciji ćemo naučiti, kako možemo programirati animacije uz pomoć kornjače.

Kako Crtati Kvadrat koji Ostavlja Tragove

U našoj prvoj animaciji, izabraćemo figuru koja nije previše komplikovana i koju znamo već odavno: Pomeraćemo jedan kvadrat s leva na desno.



Program koji crta kvadrat, znamo već od ranije:

```
to KVADRAT100
repeat 4 [fd 100 rt 90]
end
```

čim je kvadrat nacrtan, pomerimo kornjašu malo na desno i onda ponovo crtamo kvadrat. To ponavljamo nekoliko puta.

U sledećem programu, crtamo 120 tih kvadrata:

```
to POKRENIKVADRAT
repeat 120 [POKRENIKVADRAT rt 90 fd 4 lt 90]
end
```

Vežba 64

Napiši programe **KVADRAT100** i **POKRENIKVADRAT** u editoru i isprobaj **POKRENIKVADRAT**. šta će biti nacrtano?

Možeš videti, da su tragovi *svih* kvadrata nacrtani. Međutim, za našu animaciju želimo da vidimo samo poslednji kvadrat i želimo da izbrišemo prethodne tragove.

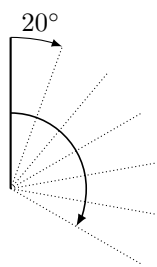


Vežba 65

Pusti kvadrat da se kreće od dole na gore umesto s leva na desno.

Vežba 66

Napiši program za liniju dužine 20, tako da se kornjača vrati na početnu tačku kad završi liniju. Koristi taj program da bi okretao tu liniju u smeru kazaljke na satu:



Kako Crtati i Opet Izbrisati Kvadrat

Da bi izbrisali tragove, moramo da naučimo kako možemo izbrisati slike koje smo skoro nacrtali. Zato kornjača mora da koristi gumicu za brisanje umesto olovke. Sa novom komandom **penerase** ili skraćeno **pe**, kornjača zamenjuje olovku sa gumicom za brisanje.

Vežba 67

Razmisli o tome šta program **KVADRAT100 pe KVADRAT100** radi , bez korišćenja kompjutera.

Da bi kornjača ponovo crtala, treba nam nova komanda: **penpaint** ili kraća verzija **ppt**. Možemo odmah koristiti novu komandu u programu iz vežbe 67. Program sad izgleda ovako:

```
KVADRAT100 pe KVADRAT100 ppt
```

Vežba 68

Isprobaj program gore. šta će se desiti? Možeš li to objasniti?

Kvadrat Mora Malo da Čeka

Kao što smo sigurno primetili u vežbi 68, kvadrat je odmah izbrisan, čim je bio nacrtan. Mi čak i ne primećujemo da je ikada bio nacrtan. Zato, pre nego što brišemo kvadrat, kompjuteru mora malo da čeka.

To možemo uraditi ovako:

wait	4
Komanda za čekanje	Vreme čekanja

Vežba 69

Isprobaj program:

```
KVADRAT100 wait 4 pe KVADRAT100 ppt
```

Kvadrat, Koji Ide s Leva na Desno

Sada smo spremni da uključimo komande za brisanje i za čekanje u naš program

Now we can include the commands for waiting and erasing the square into our program **POKRENIKVADRAT**:

```
to POKRENIKVADRAT
repeat 120 [KVADRAT100 wait 4 pe KVADRAT100 rt 90 fd 4 lt 90 ppt]
end
```

Isprobaj to. Ako ti kornjača smeta u animaciji, onda započni program sa komandom **hideturtle** (ili kraće: **ht**), koja će sakriti kornjaču. Primetićeš da je animacija sada brža. Završi program sa komandom **showturtle** (ili kraće: **st**), neposredno pre komande **end**, što će ponovo pokazati kornjaču.

Vežba 70

Pokreni kvadrat veličine 50×50 na gore.

Vežba 71

Promeni program **POKRENIKVADRAT** tako da se kvadrat kreće sa duplom brzinom nego malo pre.

Vežba 72

Da li znaš kako da promeniš program **POKRENIKVADRAT** tako da se kornjača kreće duplo sporije na desno?

Vežba 73

Promeni program **POKRENIKVADRAT** tako da se kornjača kreće od desno na levo.

Vežba 74

Prvo razmisli o tome šta će sledeći program uraditi, i onda proveri tvoju pretpostavku izvršavanjem programa:

```
to POKRENIKVADRAT1
ht
repeat 50 [KVADRAT100 wait 5 pe KVADRAT100 fd 3 rt 90 fd 3 lt 90 ppt]
KVADRAT100
st
end
```

Vežba 75

Prvo razmisli o tome šta će sledeći program uraditi. Onda proveri tvoju pretpostavku kompjuterom.

```
to KRUZITI
ht
repeat 360 [KVADRAT100 wait 4 pe KVADRAT100 fd 5 rt 1 ppt]
KVADRAT100
st
end
```

Vežba 76

Promeni program **KRUZITI** tako da bi se kvadrat okretao četiri puta brže.

Vežba 77

šta će sledeći program uraditi?

```
repeat 6 [KRUZITI]
```

Vežba 78

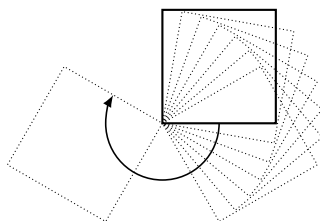
Pogledaj sledeći program

```
to ZEMLJA  
repeat 45 [fd 16 rt 8]  
end
```

i koristi ga za animaciju, u kojoj će se zemlja okretati oko sunca. Upotrebi fantaziju za prikazivanje sunca.

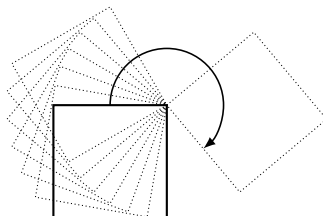
Vežba 79

Okreni jedan kvadrat u pravcu kazaljke na satu oko svog levog donjeg ćoška. Možeš izabrati veličinu kvadrata po tvojoj želji:



Vežba 80

Sada okreni kvadrat u pravcu kazaljke na satu oko svog gornjeg desnog ćoška:



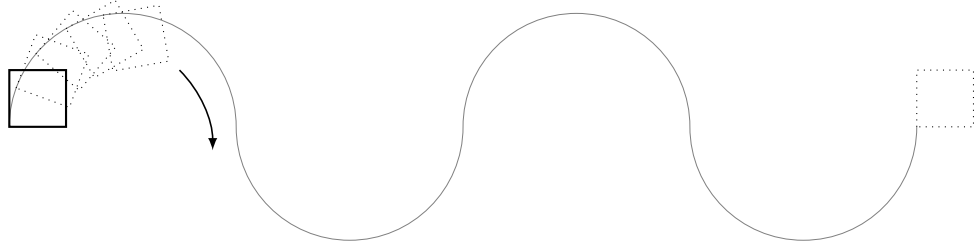
Ako već znaš šta su parametri iz lekcije 5, možeš raditi na sledećim zadacima.

Vežba 81

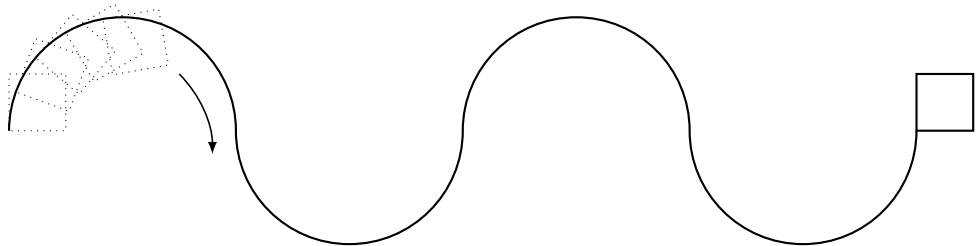
Napiši program sa *dva parametra* u kojem će se kvadrat kretati s leva na desno. Jedan parametar će odrediti veličinu kvadrata, a drugi će odrediti brzinu kvadrata.

Vežba 82

- (a) Pusti kvadrat da se šeta po dole nacrtaoj stazi, koja se sastoji od četiri polukruga. Veličina kvadrata se treba odrediti jednim parametrom.

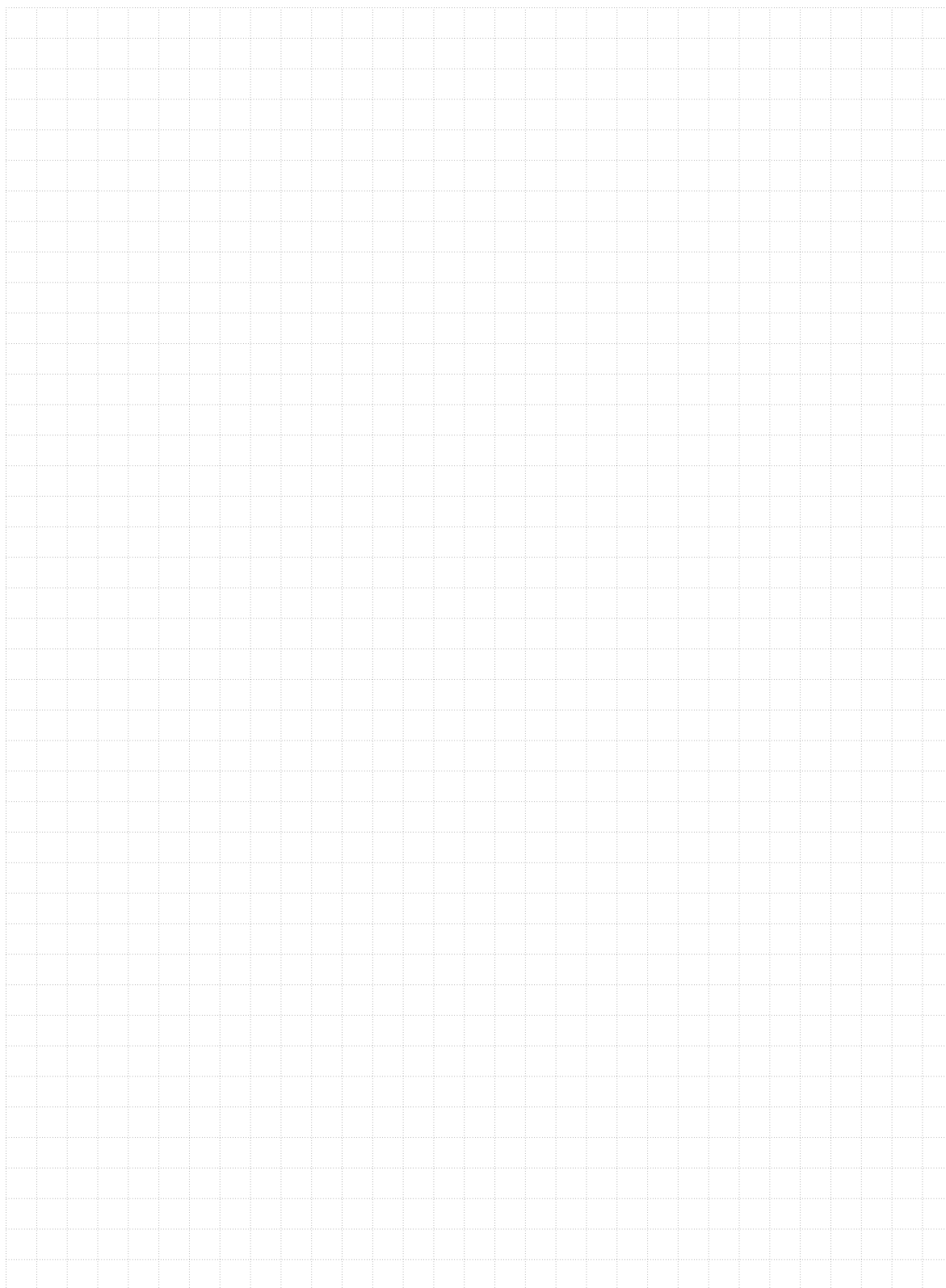


- (b) Sada takođe hoćemo da nacrtamo stazu kojom se kvadrat kreće.



- (c) Da li možeš da proširis program iz (b) tako da broj polukrugova može da se podesi jednim parametrom?

Moje beleške



Pregled komandi

- fd 100** idi 100 koraka unapred
- bk 50** idi 50 koraka unazad
- cs** izbriši sve i počni ispočetka
- rt 90** rotiraj 90 stepena na desno
- lt 90** rotiraj 90 dstepena na levo
- repeat 4 [...]** program u [...] se ponavlja četiri puta
 - pu** kornjača počinje režim hodanja
 - pd** kornjača se vraća u režim crtanja
- setpc 3** promeni boju olovke u boju 3
- to IME** stvori program sa imenom
- to IME :PARAMETAR** stvori program sa imenom i parametrom
 - end** svi programi sa jednim imenom se završavaju tom komandom
 - pe** kornjača počinje režim brisanja
 - ppt** kornjača se vraća u režim crtanja iz režima brisanja
 - wait 5** kornjača čeka 5 jedinice vremena



Programiranje sa LOGO

Informationstechnologie und Ausbildung
ETH Zurich, CAB F 15.1
Universitätstrasse 6
CH-8092 Zurich
Switzerland

www.ite.ethz.ch
www.abz.inf.ethz.ch