

1. Napiši program koji će kornjači reći da iscrta šahovsku tablu.

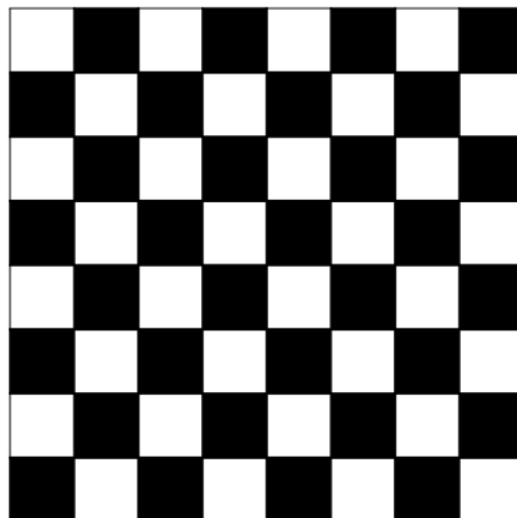
Rešenje 1:

```
import turtle
turtle.color("black")
turtle.speed(0)
duzina_polja = 50

for n in range (4):

    for m in range (2):
        for i in range (4):
            for j in range (4):
                turtle.forward(duzina_polja)
                turtle.left(90)
                turtle.forward(duzina_polja)
                turtle.begin_fill()
                for i in range (4):
                    turtle.forward(duzina_polja)
                    turtle.left(90)
                turtle.end_fill()
                turtle.forward(duzina_polja)
            turtle.left(180)

        turtle.right(90)
        turtle.penup()
        turtle.forward(2*duzina_polja)
        turtle.pendown()
        turtle.left(90)
```



Rešenje 2:

```
import turtle
turtle.color("black")
turtle.speed(0)
duzina_polja = 50

for i in range (8):
    for j in range (8):
        if ((i+j) % 2 == 1 ):
            turtle.begin_fill()
            for k in range (4): # kvadrat
                turtle.forward(duzina_polja)
                turtle.left(90)
            turtle.forward(duzina_polja)
            if ((i+j) % 2 == 1):
                turtle.end_fill()
```

```

# vrati se na pocetak sledeceg reda
turtle.left(180)
turtle.forward(8*duzina_polja)
turtle.left(90)
turtle.penup()
turtle.forward(duzina_polja)
turtle.pendown()
turtle.left(90)

```

2. Za uneto n iscrtati:

```

import turtle
turtle.color("purple")
turtle.shape("turtle")
turtle.speed(0)

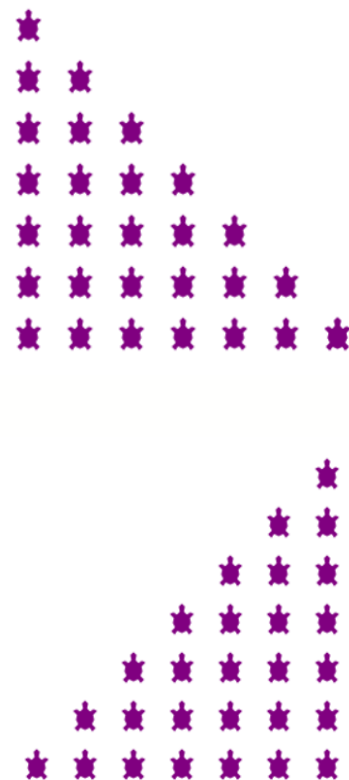
turtle.stamp

n = int(input('Unesi vrednost za n'))
d = 40

turtle.left(90)
turtle.penup()
for i in range(1,n+1):
    suma = 0
    for j in range(i):
        turtle.stamp()
        turtle.forward(d)
        suma = suma + d
    turtle.forward(-suma)
    turtle.right(90)
    turtle.forward(-d)
    turtle.left(90)

turtle.right(90)
turtle.forward(n*d)
turtle.left(90)

```



3. Napisati program koji će kornjači reći da iscrta isprekidanu liniju jednakih dužina segmenata kao što je prikazano na slici.

```
# Prvi nacin
import turtle

duzina_staze = int(input("Unesite ukupnu duzinu staze"))
duzina_segmenta = int(input("Unesite duzinu segmenta"))
predjeni_put = 0
while predjeni_put + 2*duzina_segmenta <= duzina_staze:
    turtle.forward(duzina_segmenta)
    turtle.penup()
    turtle.forward(duzina_segmenta)
    turtle.pendown()
    predjeni_put = predjeni_put+2*duzina_segmenta
if predjeni_put + duzina_segmenta > duzina_staze:
    turtle.forward(duzina_staze - predjeni_put)
else:
    turtle.forward(duzina_segmenta)
    turtle.penup()
    predjeni_put = predjeni_put + duzina_segmenta
    turtle.forward(duzina_staze - predjeni_put)
```

```
# drugi nacin
import turtle

duzina_staze = int(input("Unesite ukupnu duzinu staze"))
duzina_segmenta = int(input("Unesite duzinu segmenta"))
predjeni_put = 0
punalinija = True
while predjeni_put + duzina_segmenta <= duzina_staze:
    turtle.forward(duzina_segmenta)
    if punalinija :
        turtle.penup()
        punalinija = False
    else:
        turtle.pendown()
        punalinija = True
    predjeni_put = predjeni_put + duzina_segmenta

turtle.forward(duzina_staze - predjeni_put)
```

```

#treci nacin
import turtle

duzina_staze = int(input("Unesite ukupnu duzinu staze"))
duzina_segmenta = int(input("Unesite duzinu segmenta"))

broj_celih = duzina_staze / (duzina_segmenta*2)
ostatak = duzina_staze % (duzina_segmenta*2)

for i in range(broj_celih):
    turtle.forward(duzina_segmenta)
    turtle.penup()
    turtle.forward(duzina_segmenta)
    turtle.pendown()

if (ostatak <= duzina_segmenta):
    turtle.forward(ostatak)
else:
    turtle.forward(duzina_segmenta)
    turtle.penup()
    turtle.forward(ostatak - duzina_segmenta)

```

```

#cetvrti nacin
import turtle

duzina_staze = int(input("Unesite ukupnu duzinu staze"))
duzina_segmenta = int(input("Unesite duzinu segmenta"))

broj_celih = duzina_staze / duzina_segmenta
ostatak = duzina_staze % duzina_segmenta
punalinija = True

for i in range(broj_celih):
    turtle.forward(duzina_segmenta)
    if punalinija :
        turtle.penup()
        punalinija = False
    else:
        turtle.pendown()
        punalinija = True
turtle.forward(ostatak)

```