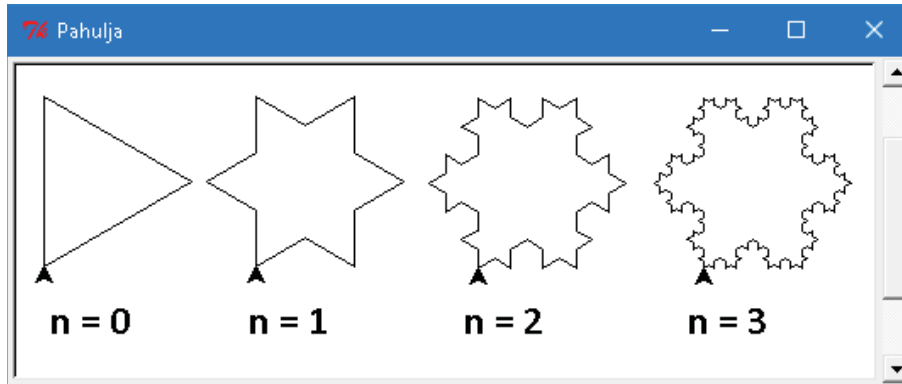


6. Rješavanje problema programiranjem

Vježba 11. Kochova pahuljica

Napišite program koji rekurzivnom procedurom pahulja(n, a) crta pahuljicu na sljedeći način:



Slika 22. Pahulja

- $n = 0$: crta se trokut stranice duljine a
- $n = 1$: svaka stranica trokuta podijeli se na četiri dijela čija je duljina $a/3$, kao na slici 23.
- $n = 2$: svaka linija nacrtana na razini $n = 1$ dijeli se na isti način itd.

Rješenje:

Promotrite rekurzivni potprogram pahulja(n, a). Za $n = 1$ nacrtana se crtež kao na shemi na slici 23.

Ako je $n > 1$, onda se za svaku ravnu crtu na slici 23. izvrši rekurzija pahulja($n-1, a/3$).

Ako je $n = 0$, crta se samo ravna crta duljine a i tu rekurzija završava.


U glavnom se programu osim upisivanja duljine stranice a i dubine rekurzije n zapravo petljom `for` crta „trokut“, samo što stranica trokuta nije ravna crta duljine a , `fd(a)`, već se poziva potprogram pahulja(n, a).

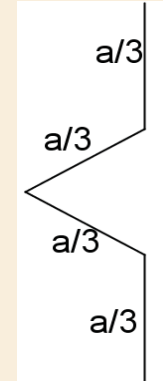
 Rješenje u datoteci: **86Bvje11.py**.

Vježba 12. Pitagorino stablo

Preko zimskih praznika učenici su bili na zimovanju. Nakon igranja na snijegu učitelj je smislio igru crtanja. On je prvi započeo igru. Nacrtao je kvadrat i nad njim jednakokrani pravokutni trokut. Papir je dao prvom učeniku koji je trebao nacrtati još jedan kvadrat i pravokutni trokut nad jednom katetom pravokutnog trokuta te dati crtež sljedećem učeniku itd. Dio crteža koji je tako nastao vidite na slici 2. Podsjeća li vas crtež na stablo?



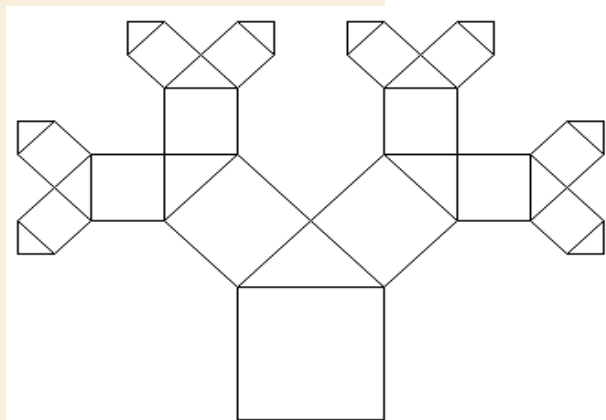
Više o fraktalima i Kochovim pahuljici pročitajte u  **86Bfraktali.pdf**.



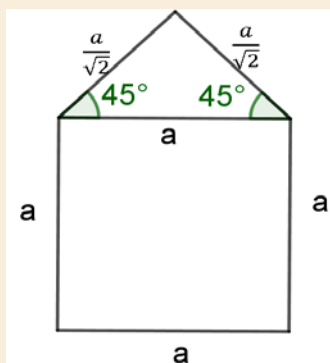
Slika 23. Shema zamjene jedne stranice

```
from turtle import *
def pahulja(n,a):
    if n == 0:
        fd(a)
    else:
        pahulja(n-1,a/3); lt(60)
        pahulja(n-1,a/3); rt(120)
        pahulja(n-1,a/3); lt(60)
        pahulja(n-1,a/3)

title('Pahulja')
lt(90)
a = textinput('Veličina','a = ')
a = int(a)
n = textinput('Dubina rekurzije','n = ')
n = int(n)
for k in range(3):
    pahulja(n,a)
    rt(120)
```



Slika 24. Stablo



Slika 26. Shema Pitagorina stabla

- Koliko je trokuta i kvadrata na slici? Morate li brojati ili postoji bolji način za određivanje broja kvadrata i trokuta?
- Crtež je simetričan. Ako se nastavi crtati, koliko najmanje djece mora crtati kvadrate i trokute da lik i dalje bude simetričan?

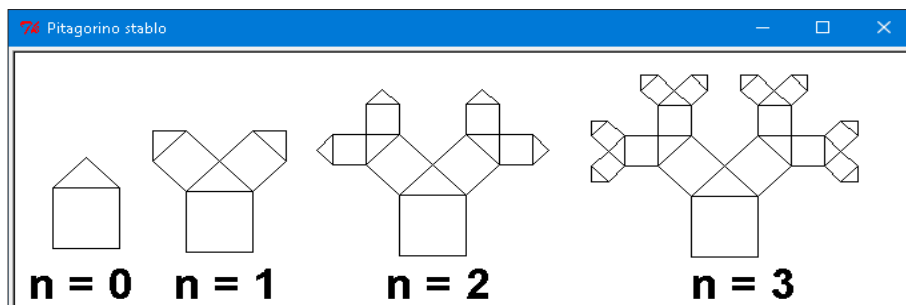
Ovo je zapravo rekurzija koju nazivamo Pitagorino stablo.

Rješenje:

Objasnimo potprogram `pitagora(n, a)`. Na razini $n = 0$ crta se kvadrat stranice a (potprogramom `kvadrat(a)`) i jednakokrani pravokutni trokut hipotenuze duljine a .

Njegove su katete duljine.

Za $n > 0$ nad svakom stranicom trokuta poziva se isti potprogram, ali jedne razine manje i kraće stranice.



Slika 25. Pitagorino stablo

```

from turtle import *
from math import *
def kvadrat(a):
    for k in range(4):
        fd(a); rt(90)

def pitagora(a,n):
    if n>=0:
        kvadrat(a);
        fd(a); lt(45);
        pitagora(a/sqrt(2),n-1)
        rt(90); fd(a/sqrt(2));
        pitagora(a/sqrt(2),n-1)
        rt(90); fd(a/sqrt(2));
        bk(a/sqrt(2)); lt(90);
        bk(a/sqrt(2)); lt(45); bk(a)

a = textinput('Duljina stranice','a = ')
a = int(a)
n = textinput('Dubina rekurzije','n = ')
n = int(n)
lt(90); title('Pitagorino stablo'); ht();
pitagora(a,n)

```

Kornjača počinje crtati iz donjeg lijevog vrha kvadrata, pa ju je važno na kraju potprograma uvijek vratiti na to mjesto, inače rekurzija neće dobro crtati.

U ovom programu treba nam ugrađena funkcija `sqrt()` za računanje drugog korijena. Zbog nje treba unijeti modul `math` koji je sadržava.

Python raspolaže naredbama koje možemo upotrijebiti odmah nakon njegovog pokretanja. Međutim, postoje i skupine naredbi koje prije upotrebe moramo učitati, odnosno unijeti u Python. Takve skupine programa zovu se **moduli**. Svaki modul ima svoje ime. Na primjer, želimo li crtati kornjačom, moramo unijeti modul `turtle`, a želimo li izvršavati složenije matematičke operacije, unijet ćemo modul `math`.



Rješenje u datoteci: **86Bvje12.py**.