



Modulska poštevanka

Modulska aritmetika imenovana tudi urna aritmetika.

Operaciji seštevanja in množenja v modulski aritmetiki računamo kot v običajni aritmetiki, le da rezultat na koncu delimo z danim številom, ki ga imenujemo *modul*, in kot rezultat privzamemo ostanek pri deljenju.

Urna aritmetika se imenuje zato, ker ure preračunavamo na enak način, po modulu 12 oziroma 24. Če je ura kaže 10, bo čez 4 ure kazala 2, ker je ostanek pri deljenju $(10+4)/12$ enak 2, $14/12 = 1 + 2/12$.

V programskem jeziku PYTHON operator % računa ostanek pri deljenju. Tako je $5\%3$ enako 2.

Definicija problema

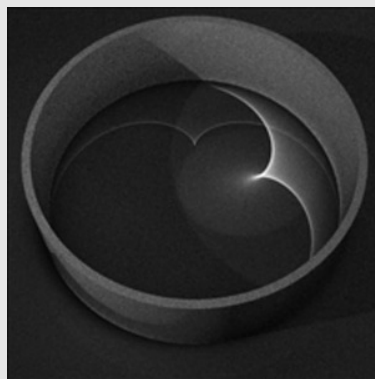
Napisali bomo program, ki bo grafično predstavil modulsko poštevanko.

Za izbran modul m bomo na krožnici označili enakomerno razporejene točke, ki so med seboj zamaknjene za $360/m$ stopinj. Točke označimo z indeksom, ki teče od 0 do $m-1$. Izberimo poštevanko, na sliki je to poštevanka števila $n = 2$. Povezujemo točko z indeksom x_1 s točko z indeksom $x_2 = n \times x_1 \bmod m$, kjer indeks x_1 teče od 0 do $m-1$.

Zanimiva je ovojnica daljic pri poštevankah. Za poštevanko $n = 2$ dobimo krivuljo, ki se imenuje *srčnica*. Za $n = 3$ pa dobimo krivuljo, ki se imenuje *nefroida ali kavstika*, ki je ovojnica odbitih vzporednih žarkov na krožnem zrcalu.

Vhod

Polmer krožnice je $R = 300$ modul $m = 180$. Poštevanko n bomo prebrali z ukazom `input`.



```
#!/usr/bin/env python3
from bokeh.plotting import figure, output_file, show
import json, math
import sys

def frange(b,e,s):
    h=[]
    x = b
    while x <= e:
        h.append(x)
        x += s
    return h

n = int(input('pošteevanka: '))
m = 360
a = 1
r = 300

h = frange(0,360,1)
x = [r*math.cos(t*math.pi/180) for t in h]
y = [r*math.sin(t*math.pi/180) for t in h]

xs = []
ys = []
for i in range(m):
    j = (i * n) % m
    xs.append([x[i],x[j]])
    ys.append([y[i],y[j]])

output_file("modulo.html")
p = figure(title="Modulska aritmetika")
p.scatter(x, y, size=2, color="#3A5785", alpha=0.6)
#p.line(x,y, color = "blue", line_width = 2)
for k in zip(xs,ys):
    p.line(*k, color="#3A0085", alpha=0.1, line_width = 1)
show(p)
```