Mandelbrotov fraktal

Marko Čupić; 2019.

Iterativno preslikavanje

- Neka je *C* neki kompleksan broj (odnosno točka u kompleksnoj ravnini)
- Promotrimo preslikavanje $z_{n+1} = z_n^2 + C$ gdje je

$$z_0 = 0 + 0i$$

$$z_1 = z_0^2 + C = C$$

$$z_2 = z_1^2 + C = C^2 + C$$

$$z_3 = z_2^2 + C = (C^2 + C)^2 + C$$

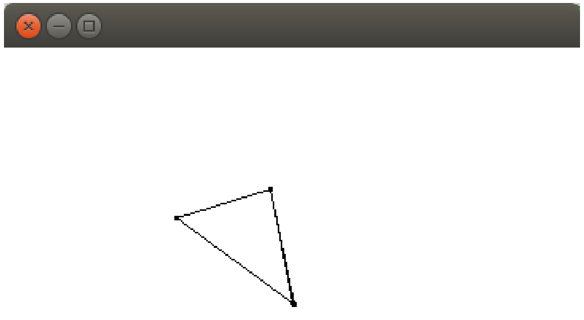
$$z_4 = z_3^2 + C = ((C^2 + C)^2 + C)^2 + C$$

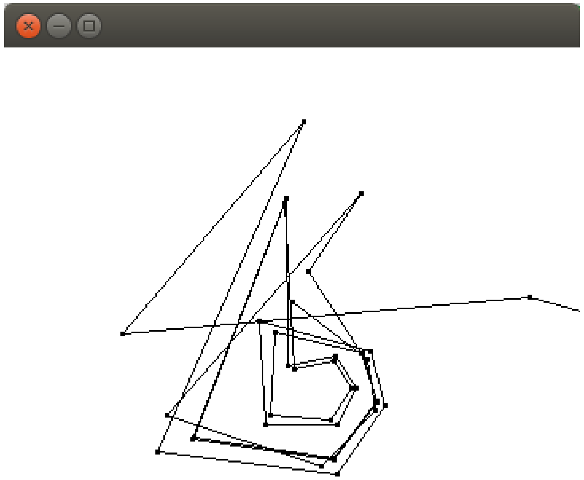
Što se događa s tim kompleksnim brojevima koje dobivamo, kako iteracije napreduju prema beskonačnosti?

Iterativno preslikavanje

- Imamo tri mogućnosti:
  - Modul točaka raste prema beskonačnosti (točke se udaljavaju od ishodišta; niz divergira)
  - Modul točaka se mijenja kroz iteracije, ali točka se nikada od ishodišta ne udaljava više od nekog konačnog praga
  - Točke konvergiraju prema nekoj fiksnoj točki (konačnoj)

Mandelbrotov skup



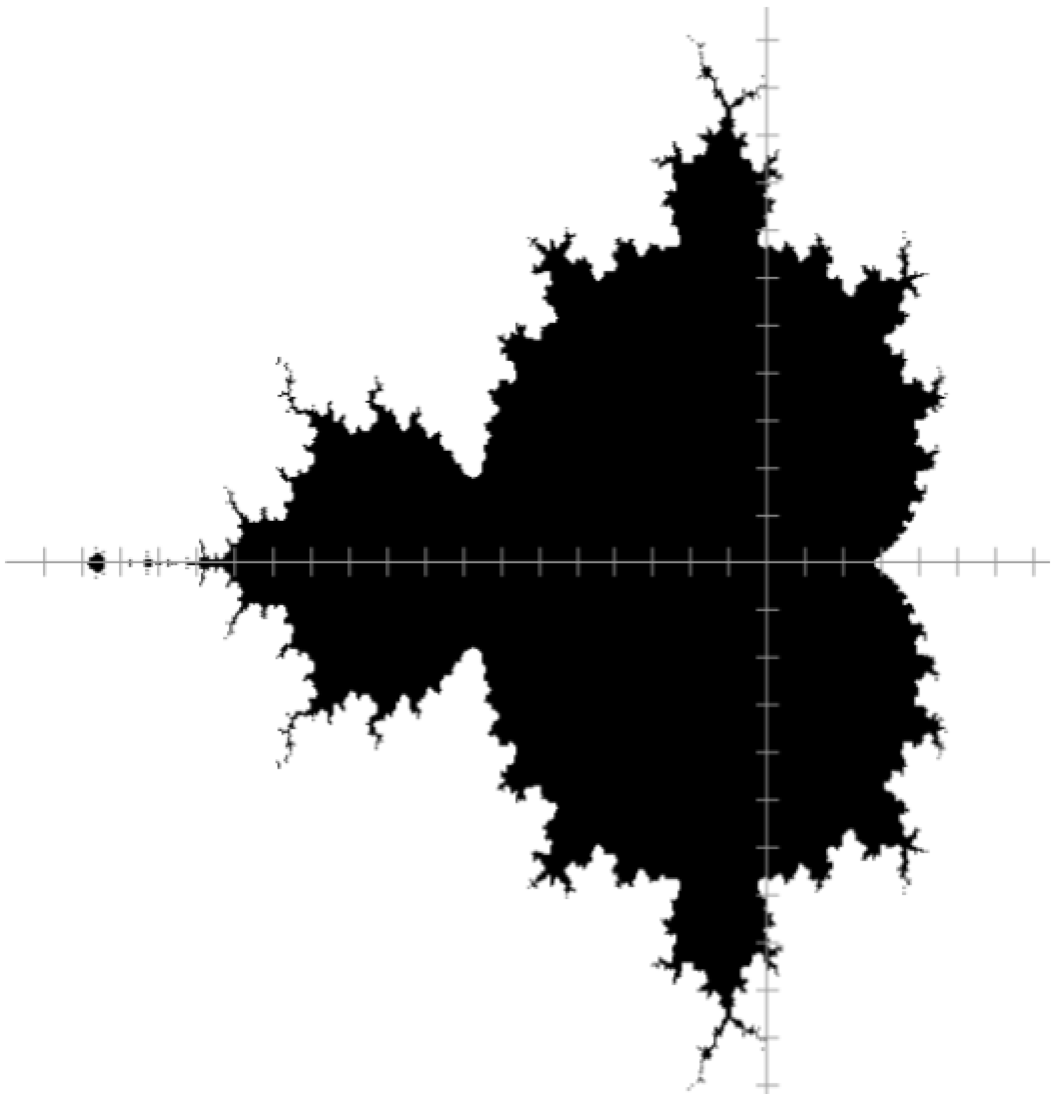


(primjer: IterationViewer)

Mandelbrotov skup

- Čitavu kompleksnu ravninu (beskonačno je velika!) možemo podijeliti u dva skupa točaka
    - Točke za koje niz divergira, ako iterativno preslikavanje pokrenemo iz njih, te
    - Točke za koje niz NE divergira, ako iterativno preslikavanje pokrenemo iz njih (obuhvaća druga dva slučaja); ove točke čine ***Mandelbrotov skup***
  - Da bismo ga ilustrirali, možemo gusto uzorkovati točke kompleksne ravnine, provesti za svaku ispitivanje, pa ako pripada skupu, obojati je crno

Mandelbrotov skup



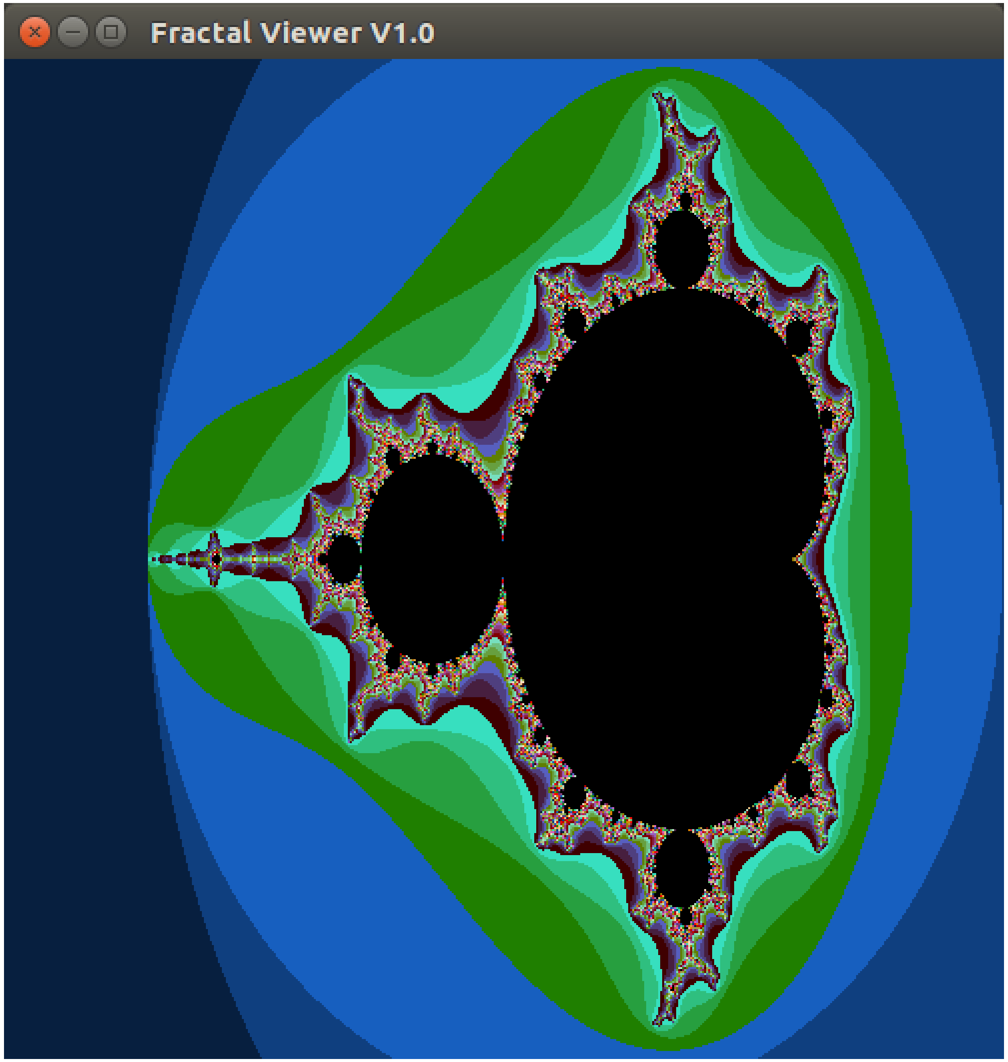
Mandelbrotov skup

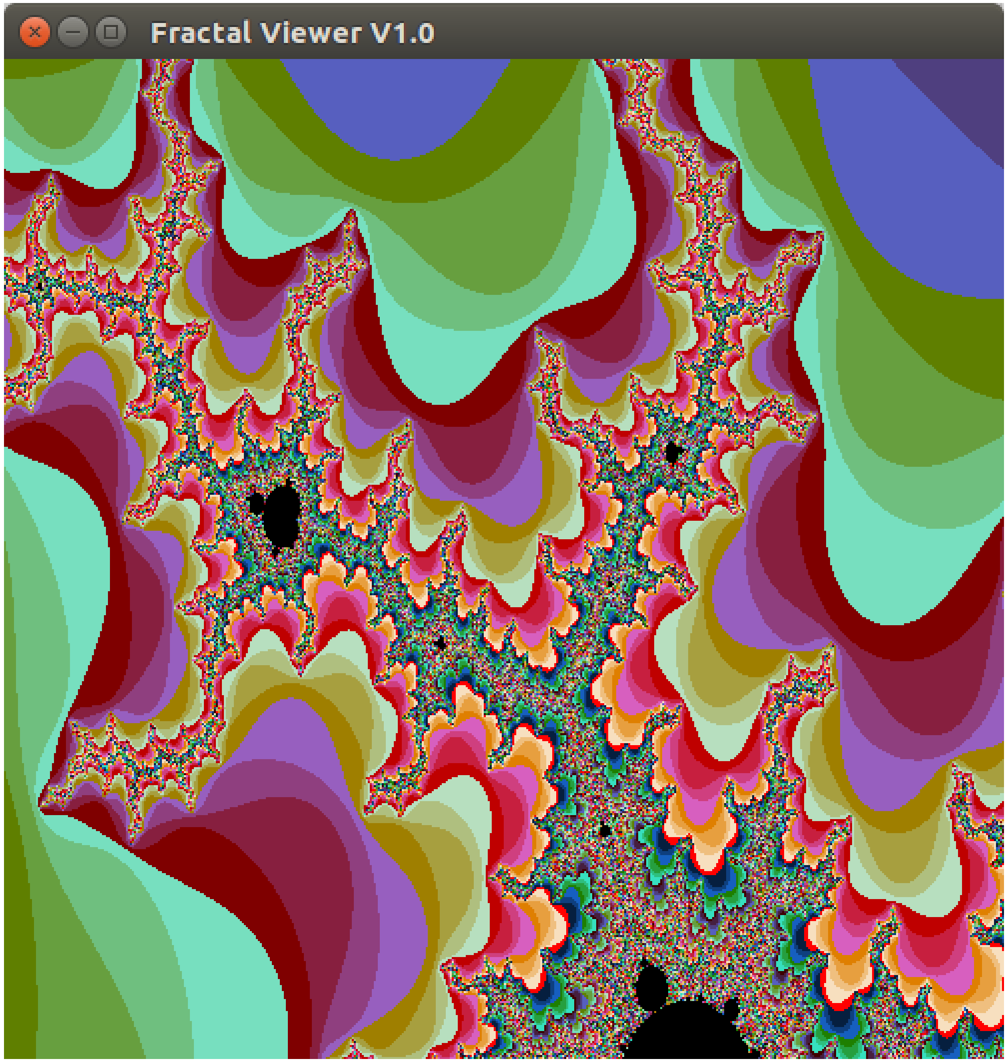
- Uočite, točaka ima beskonačno, i ispitivanje za svaku koja ne divergira traje beskonačno
- U programskoj implementaciji, morat ćemo postaviti neke ograde
  - Iz matematike znamo da će ovo preslikavanje divergirati ako bilo koja dobivena točka po modulu postane veća od 2 – to ćemo koristiti kao jedan uvjet
  - Ako ne utvrdimo divergenciju *unutar zadanog broja iteracija*, pretpostavit ćemo da ne divergira

Mandelbrotov fraktal

- ***Mandelbrotov fraktal*** je granica Mandelbrotovog skupa (na prethodnom slideu granica između crnoga i bijeloga)
- Često se, međutim, pod ovim imenom podrazumijeva slika koja različim bojama boja točke ovisno o tome koliko brzo divergiraju (tj. koliko nam iteracija treba da bismo došli do točke čiji je modul postao veći od zadane granice)

Mandelbrotov fraktal





Mandelbrotov fraktal

- Kako točno crtamo? Očito ne možemo promatrati čitavu kompleksnu ravninu!
  - Pretpostavimo da gledamo dio kompleksne ravnine čiji je realni dio između u_{min} i u_{max} a imaginarni dio između v_{min} i v_{max}.
  - Iznad tog dijela postavite raster zaslona koji je $w \times h$ slikovnih elemenata; (0,0) “lebdi” iznad (u_{min},v_{max}) u kompleksnoj ravnini; (w-1,h-1) iznad (u_{max},v_{max}) (naime, ekranski koordinatni sustav ima ishodište gore lijevo)

Mandelbrotov fraktal

- Sada je dalje lagano: umjesto beskonačno točaka u kompleksnoj ravnini, moramo samo istražiti ponašanje ispod $w \times h$ točaka (za svaki slikovni element odredimo točku kompleksne ravnine iznad koje smo ga postavili; za tu točku ispitamo ponašanje niza; obojamo slikovni element)
- Imamo li raster 1000×1000, to je 10^6 pokretanja algoritma koji i sam može trajati
  => paralelizacija

Mandelbrotov fraktal

- Ako se slikovni element nalazi na koordinatama (x,y), ispod njega je točka kompleksne ravnine:

$$c_{\mathrm{re}}=\frac{x}{w-1}\cdot\left(u_{max}-u_{min}\right)+u_{min}$$

$$c_{\mathrm{im}}=\frac{h-1-y}{h-1}\cdot\left(v_{max}-v_{min}\right)+v_{min}$$

Mandelbrotov fraktal

- Izračun

$$z^2 + c = (z_{\mathrm{re}} + i \cdot z_{\mathrm{im}})^2 + c_{\mathrm{re}} + i \cdot c_{\mathrm{im}}$$
$$= z_{\mathrm{re}}^2 + 2 z_{\mathrm{re}} z_{\mathrm{im}} i - z_{\mathrm{im}}^2 + c_{\mathrm{re}} + i \cdot c_{\mathrm{im}}$$
$$= (z_{\mathrm{re}}^2 - z_{\mathrm{im}}^2 + c_{\mathrm{re}}) + i \cdot (2 z_{\mathrm{re}} z_{\mathrm{im}} + c_{\mathrm{im}})$$

Mandelbrotov fraktal

- Pripremljena je metoda:

```
Mandelbrot.calculate(
    double reMin, double reMax,
    double imMin, double imMax,
    int width, int height, int m,
    int ymin, int ymax,
    short[] data, AtomicBoolean cancel
)
```