DATALOGI A
Projekt C

Dette projekt er et af fem obligatoriske projekter der tilsammen udgør eksamen i Datalogi A. Projektet besvares i grupper på 2; eventuelt efter foregående aftale med instruktor i grupper på 1 eller 3.

Projektet afleveres senest mandag den **3. november** klokken **12**.

Baggrund

De såkaldte *Julia-mængder* er delmængder af planen $\mathbb{R}^2$ defineret ved visse funktioners iterative opførsel. Man betragter for eksempel $f : \mathbb{R}^2 \to \mathbb{R}^2$ givet ved

$$f(x, y) = (x^2 - y^2, 2xy + 1)$$

og udregner for et givet par af koordinater (x_0, y_0) talparrene

$$f^{\circ n}(x_0, y_0),$$

hvor $f^{\circ n}(x_0, y_0)$ er resultatet af at tage f n gange på (x_0, y_0), altså

$$\begin{aligned}
f^{\circ 1}(x_0, y_0) &= f(x_0, y_0) \\
f^{\circ 2}(x_0, y_0) &= f(f(x_0, y_0)) \\
f^{\circ 3}(x_0, y_0) &= f(f(f(x_0, y_0)))
\end{aligned}$$

og så videre. Vi sætter $f^{\circ 0}(x_0, y_0) = (x_0, y_0)$.

Hvis

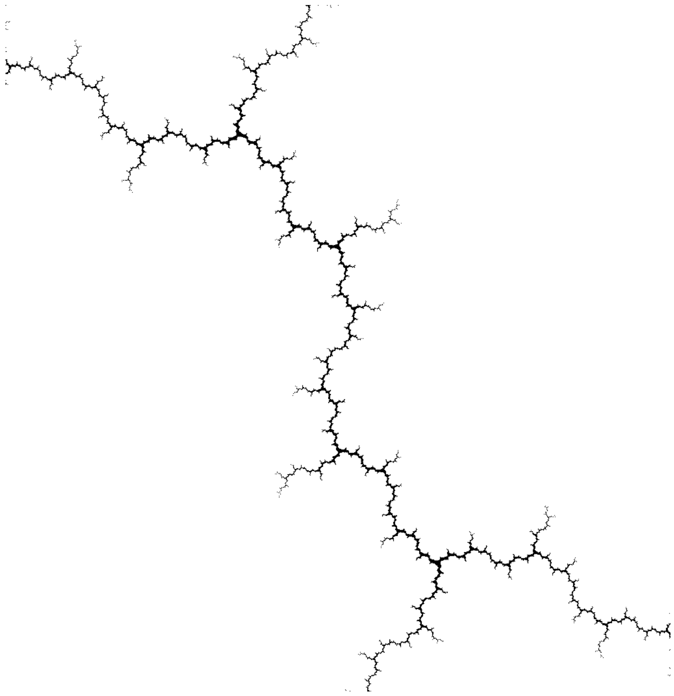
$$|f^{\circ n}(x_0, y_0)|^2 \leq \tfrac{3}{2} + \tfrac{1}{2}\sqrt{5}$$

for alle n siger man at (x_0, y_0) ligger i Julia-mængden. Her er

$$|(x, y)|^2 = x^2 + y^2,$$

så spørgsmålet er altså om kvadratet på længden af $f^{\circ n}(x_0, y_0)$ holder sig under $\frac{3}{2} + \frac{1}{2}\sqrt{5} \simeq 2.618033990$.

I praksis beslutter man sig for et maksimalt antal iterationer $max_{iteration}$ og sætter en sort prik i koordinatsystemet ved (x_0, y_0) hvis $|f^{\circ n}(x_0, y_0)|^2 \leq 2.7$ for alle n fra 0 op til $max_{iteration}$. Det er sådan Figur 1 er fremkommet.



Figur 1: Juliamængde [$4 \cdot 10^6$ punkter, $max_{iteration} = 17$]

Det er et generelt fænomen at jo større $max_{iteration}$ vælges, jo flere pukter må man tage under behandling for at få figuren frem. Det nødvendiggør mange udregninger, og derfor er det beregningsmæssigt tungt at tegne Julia-mængder præcist op. Vi skal se på hvor meget hurtigere det går at gøre den slags i C end i Maple.

Opgavetekst

1. Overbevis dig om at pseudokoden

```
Givet x0, y0 og maxiteration

Sæt tæller til 0
Sæt (x,y) lig (x0,y0)

Mens kvadratet på længden af (x,y) er mindre end eller lig 2.7 og
                            tæller er mindre end maxiteration
        Sæt (x1,y1) lig f(x,y)
        Sæt (x,y) lig (x1,y1)
        Læg 1 til tæller
```

```
Hvis kvadratet på længden af (x,y) er mindre end eller lig 2.7
        returner 1
ellers
        returner 0
```

vil returnere 1 hvis der gælder $|f^{\circ n}(x_0, y_0)|^2 \leq 2.7$ for alle n fra 0 op til $max_{iteration}$, og returnere 0 ellers.

2. Omsæt pseudokoden til en C funktion

```
int julia(double x0, double y0, int maxiteration)
{

}
```

Oversæt din funktion sammen med det udleverede program

```
L:/dat-a/projekter/projektc/projektc.c
```

(vedlagt udskrift i appendiks) og kør det. Hvis ikke resultaterne passer med de udskrevne facit må der være en fejl i dit program – vend tilbage og ret den!

3. Programmet `projektc.c` måler også tidsforbruget i sekunder for at udregne funktionen `julia` 10000 gange med de givne parametre. Hertil benyttes funktionerne `taend_ur()`, `sluk_ur()` og `vis_ur()`, der fungerer på den måde at hvis man i C programmet skriver

```
taend_ur();
... nogle C kommandoer ...
sluk_ur();
```

så måles tidsforbruget af kommandoerne i sekunder og kan udskrives ved

```
printf("[TID = %f]", vis_ur());
```

Undersøg på denne måde tidsforbruget ved at

(a) Udregne `julia(0.0, n*0.0001, 25)` for alle heltal n mellem -10000 og 10000.

(b) Udregne `julia(0.3, n*0.0001, 25)` for alle heltal n mellem -10000 og 10000.

(c) Udregne `julia(0.6, n*0.0001, 25)` for alle heltal n mellem -10000 og 10000.

De tilsvarende tidsforbrug i Maple (på BOHR; den maskine vi benytter ved forelæsningerne) ses i Figur 2.

(a)	266.912 s
(b)	182.844 s
(c)	137.237 s

Figur 2: Tidsforbrug i Maple

Aflevering

Projektet besvares med et C program `projektc.c` der afleveres elektronisk. Filen skal ligge i et underkatalog ved navn `projektc` i en af projektgruppens medlemmers hjemmekatalog ved afleveringen.

Programmet, der baseres på det udleverede version af `projektc.c`, skal kunne oversættes og køres fra Turbo C++. Programmet skal indeholde gruppens definition af funktionen `julia` og de kommandoer der er benyttet ved tidstagningen i opgave 3. Endvidere skal resultaterne af tidskørslerne indføres som kommentarer i programmet.

Den elektroniske aflevering af projektet må foregå fra datamaten `Arnold`. Vælg ikonen `Login arnold` på Windows maskinerne i A 108 og indtast brugernavn og password for det af medlemmerne i projektgruppen i hvis hjemmekatalog besvarelsen ligger. Tast herefter

```
aflever
```

og følg dialogen. Aflevering kan ske fra mandag den 27. oktober.