

# Opgave 1: Scankonvertering af Linier

Knud Henriksen  
Department of Computer Science  
University of Copenhagen<sup>†</sup>

Fri 25<sup>th</sup> Nov, 2016

Denne opgave går ud på at forklare og implementere den algoritme til skankonvertering af rette linier, som blev gennemgået ved forelæsningen.

Afleveringen bør indeholde:

- Beskrivelse af problemet.
- En forklaring af algoritmen, som er tilstrækkelig detaljeret til at overbevise læseren om, at du har forstået algoritmen.
- En beskrivelse af eventuelle specialtilfælde.
- En implementation af algoritmen i C++.
- En afprøvning, der viser, at algoritmen tegner de rigtige pixels, samt at den kan tegne linier med hældninger i alle 8 oktanter. Der er udleveret et programskelet, som kan hjælpe med dette. At bruge billeder til at illustrere, at algoritmen virker korrekt, er den nemmeste måde at dokumentere dette.

Der er lagt et programskelet op på Absalon, så det skulle være lettere at komme i gang.. Man behøver ikke bruge dette programskelet, hvis man hellere vil lave sit eget program.

For at tegne store fede punkter, så findes der et fragmentprogram der kan dette på følgende link:

<http://mmmovania.blogspot.dk/2010/12/circular-point-sprites-in-opengl-33.html>

Hvis I benytter dette link, så husk, at I ved initialiseringen af OpenGL skal kalde:

```
glEnable(GL_PROGRAM_POINT_SIZE);  
glPointSize(floating point number); // e.g. 10.0f
```

ellers virker det sikkert ikke.

Tip! Skankonverter linier, der starter i  $(0,0)$  og ender med  $x, y \in [1, 9]$ , og efter I har skankonverteret linien, hvor I udelukkende bruger heltal, så gem dem i en buffer, f.eks.

---

<sup>†</sup>Address: Universitetsparken 5, DK-2100 Copenhagen Ø, Denmark

```

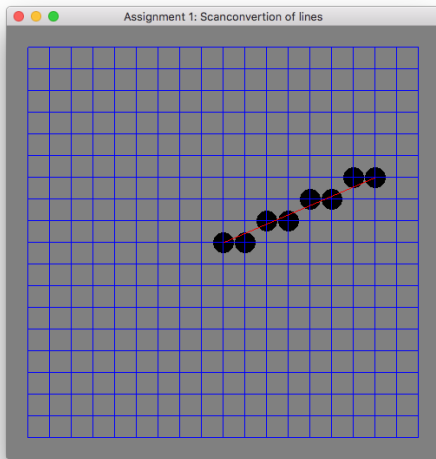
#include <vector>
...
std::vector<glm::vec3> points;
...
// beregn de heltallige koordinater (x, y) for Jeres pixel - se slides
...
points.push_back(0.1f * glm::vec(x, y, 0));
...
// de ligger nu i points og der er points.size() af dem, aom alle er type glm::vec3
...
// når I tegner næste linie, så husk at udføre
points.clear();
// for at slette de tidligere data

```

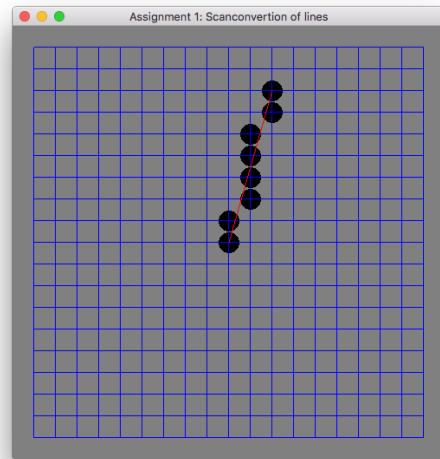
Det er Jeres data, og de skal så gives til grafikkortet, som beskrevet i mine slides.

Husk! at I skal dokumentere Jeres aflevering med billeder (screenshots), men I skal også vise, at I har forstået algoritmen ved hjælp af matematik i egne ord, og ikke bare mine slides! Mine slides er en hjælp til Jer; men det er *ikke* Jeres noteapparat (det er nemlig Jeres egne notater).

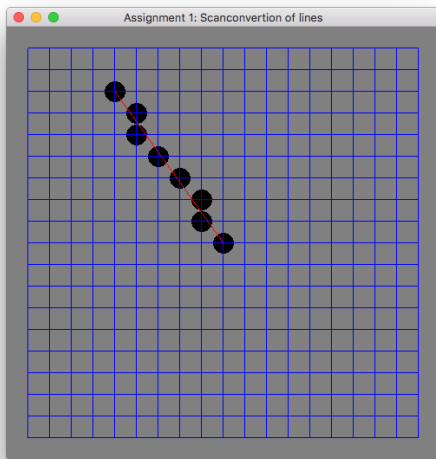
Der er også lagt et skrift op på Absalon med navnet ReportGuide.pdf. Dette skrift giver nogle gode råd om, hvad man bør have med i sin aflevering, og også et par ting, som man skal undgå.



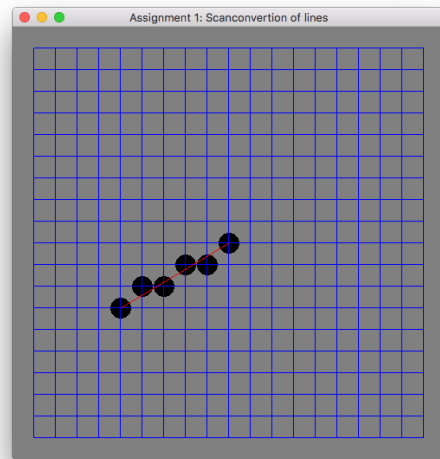
(a) En linie med negativ hældning og  $\alpha < 1$ .



(b) En linie med negativ hældning og  $\alpha > 1$ .



(c) En linie med negativ hældning og  $|\alpha| > 1$ .



(d) En linie med negativ hældning og  $|\alpha| < 1$ .