

Implementatieplan ImageShell

Vision

Stefan van der Ham & Bas van Rossem

1. Implementatieplan ImageShell

1.1. Namen en datum

Stefan van der Ham & Bas van Rossem

14-4-2019

1.2. Doel

De RGBImageStudent en de IntensityImageStudent klasse implementeren

1.3. Methoden

- 1D array
- 2D array
- 1D vector
- 2D vector

<https://en.cppreference.com/w/cpp/container/vector>

<https://en.cppreference.com/w/cpp/language/array>

Een vector is een dynamische container die zijn informatie op de heap opslaat. doordat deze dynamisch is. is hij ook trager omdat als hij groeit moet hij kopieën gaan maken wat lang duurt. Een array daarin tegen wordt op de stack opgeslagen en is statisch. Zijn grootte is dus van te voren al bepaald en gereserveerd op het systeem. Het grootte verschil tussen de twee is vooral snelheid.

<https://pediaa.com/what-is-the-difference-between-1d-and-2d-array/>

dan hebben we nog het verschil tussen 1d en 2d.

het verschil tussen 1d en 2d is de structurering van data.

1d is eigenlijk gewoon een lijst.

2d is eigenlijk net als een Excel sheet (met kolommen en rijen)

1.4. Keuze

We gaan voor een array, aangezien snelheid erg belangrijk voor ons is. En we weten de grootte van het plaatje van te voren.

We kiezen voor een 1d array omdat wij dit een fijnere manier van data structureren vinden

1.5. Implementatie

Wij implementeren de functies uit de IntensityImageStudent.h en RGBImageStudent.h daarnaast voegen wij bij allebei een array toe met de naam image van het type van de image. voor de functie: void setPixel(int x, int y, Intensity pixel); gebruiken we de volgende formule:

```
pixel_index = y * Width + x;  
Image[pixel_index] = pixel;
```

1.6. Evaluatie

Om te testen of onze implementatie werkt runnen wij de Gui software. En dan vergelijken wij het resultaat met de default implementatie.

Om te testen of onze implementatie sneller is. Maken wij gebruik van de vision timer (<https://github.com/arnokamphuis/vision-timer>)

Om te testen of onze implementatie efficiënter is. Maken wij gebruik van de Visual studio memory snapshot functie.