

# RGB

Jon Sporring

September 26, 2019

## 1 Lærervejledningn

**Emne** Højere-ordens funktioner, currying

**Sværhedsgrad** Middel

## 2 Introduktion

En farve repræsenteres ofte som en triple (rød, grøn, blå), hvor hver indgang kaldes en farvekanal eller blot en kanal, og hver kanal er typisk et heltal mellem 0 og 255:

$$c = (r, g, b) \tag{1}$$

Farver kan lægges sammen ved at addere deres kanaler,

$$c_1 + c_2 = (\text{trunc}(r_1 + r_2), \text{trunc}(g_1 + g_2), \text{trunc}(b_1 + b_2)), \tag{2}$$

$$c_i = (r_i, g_i, b_i) \tag{3}$$

$$\text{trunc}(v) = \begin{cases} 0, & v < 0 \\ 255, & v > 255 \\ v, & \text{ellers} \end{cases} \tag{4}$$

og farver kan skaleres ved at gange hver kanal med samme konstant.

$$ac = (\text{trunc}(ar), \text{trunc}(ag), \text{trunc}(ab)) \tag{5}$$

Farver, hvor alle kanaler har samme værdi,  $v = r = g = b$ , kaldes gråtoner, og man kan konvertere en farve til gråtone ved at udregne gråtoneværdien som gennemsnittet af de 3 kanaler,

$$v = \text{gray}(c) = \frac{r + g + b}{3} \tag{6}$$

### 3 Opgave(r)

1. Skriv en signaturfil for et modul, som indeholder funktionerne `trunc`, `add`, `scale`, og `gray` ud fra ovenstående matematiske definitioner og ved brug af tupler, hvor muligt.
2. Skriv en implementation af ovenstående signaturfil og kompilér begge filer til et bibliotek (dll-fil).
3. Skriv 2 programmer: Et som benytter ovenstående bibliotek via `fsharpi` og et som benytter det via `fsharpc`.
4. Lav en White-box afprøvning af jeres bibliotek og ved brug af `fsharpc`.
5. Udvid biblioteket (både signatur og implementationsfilen) med en funktion, som konverterer en farvetriplet til en gråtonetriplet. Udvid afprøvningen med en passende afprøvning af den nye funktion. Diskutér om bibliotek, program, og afprøvning er struktureret på en måde, så denne udvidelse har været let, eller om der er u hensigtsmæssige afhængigheder, som gør rettelse, vedligeholdelse og udvidelse besværlig og med risiko for fejl.