Programmering B Rapport

ELEV: Nelisa Chehrehgosha, Mie Norre Engemann

FAG: Programmering B

KLASSE: 2.R

SKOLE: Aarhus Gymnasium

VEJLEDER: Mirsad Kadribasic

DATO FOR AFLEVERING: 1/05/2022

Indholdsfortegnelse

[1 Abstract 3](#_Toc102227843)

[2 Problemforulering 3](#_Toc102227844)

[3 Programmet 3](#_Toc102227845)

[library 3](#_Toc102227846)

[3.1 design 5](#_Toc102227847)

[3.1.1 flowchart 7](#_Toc102227848)

[3.1.2 pseudokode 8](#_Toc102227849)

[3.1.3 GUI formelknapper 8](#_Toc102227850)

[3.1.4 scrum 8](#_Toc102227851)

[3.2 dokumentation 8](#_Toc102227852)

[3.2.1 kode kommentar 8](#_Toc102227853)

[3.2.2 kode stil 8](#_Toc102227854)

[3.2.3 funktion beskrivelser 8](#_Toc102227855)

[3.2.4 klasse beskrivelse og diagram 8](#_Toc102227856)

[3.3 deling 8](#_Toc102227857)

[3.3.1 github 9](#_Toc102227858)

[4 test 10](#_Toc102227859)

[Code kommentar 10](#_Toc102227860)

[Code style 11](#_Toc102227861)

[funktion beskrivelser 12](#_Toc102227862)

[class beskrivelse og diagram 14](#_Toc102227863)

[Konklusion 15](#_Toc102227864)

[Bilag 15](#_Toc102227865)

# 1 Abstract

# 2 Problemforulering

# 3 Programmet

## library

## 

## 3.1 design

## 3.1.1 flowchart

Diagram

Description automatically generated  
Figur 1. Flowchart af project (macro)

## 3.1.2 pseudokode

## 3.1.3 GUI formelknapper

## 3.1.4 scrum

Her er vores tidsplan over formelregner lavet med Trello.

<https://trello.com/b/XEB5Xn0L/formelregner>

Graphical user interface, application

Description automatically generatedFigur 2. Billede af vores tidplan fra Trello.

## 3.2 dokumentation

## 3.2.1 kode kommentar

Til venstre er et eksempel på, hvordan vi har brugt kommentarer i dette projekt til at informerer om, hvad de forskellige kommandoer gør. Når man samarbejder på et programmeringsprojekt, er det ofte forvirrende og måske endda kaotisk at se på et stykke kode for første gang som en anden har person har skrevet. Det er derfor vi bruger kommentarer.

Text

Description automatically generatedI gennem vores kode vi har brugt kommentarerne til at forklarer hvad bestemte linjer eller funktioner gør, og hvilken rolle de spiller i vores projekt. De forklarer nogle gange konkret hvad en linje bestemmer, men andre gange giver de bare et overblik over et interval eller forskellige variable som har betydning for udkommet.  
Figur 3. Billede af del af klassen Button

Det meste af tiden har vi lavet en kommentar øverst i koden, hvor vi forklarer, hvad hele denne if-sætning handler om, og hvad den bruges til.

Til sidst har vi også nogle enkelte steder udkommenteret flere linjer kode, fordi vi ikke skulle bruge dem i det øjeblik, men de kunne være gode at have senere, hvis man for eksempel ville ændre noget.

## 3.2.2 kode stil

Text

Description automatically generatedTil venstre er et eksempel på, hvordan vi har brugt kommentarer i dette projekt til at informerer om, hvad de forskellige.

Det gjorde vi både i vores klasse og i sketch fillen. Dette hjælper os med at finde det vi ønsker at rette eller forbedre senere.

Figur 4. Billede af Sketch i p5

## 3.2.3 funktion beskrivelser

Paint programmet benytter sig af flere funktioner. En af funktionerne i klassen ’Button’ hedder ’knap’ og bruges bl.a. til at lave vores GUI (*graphical user interfave*) som ses ved programmets knapper.

Når musen holdes over en knap, så vil knappens farve blive lidt mørkere. Bliver der trykket på knappen, så vil der tegnes en endnu mørkere rektangel ovenpå, som så vil forsvinde igen ved et nyt tryk på samme knap.  
Netop den sidste del med hvad der sker når man trykker på en knap, handler nedenstående Nassi Schneider’s funktions diagram og flowchart om.

Diagram

Description automatically generated  
Figur 5. Nassi Schneider funktions diagram over hvad  
der sker når der trykkes på en knap i funktionen ’knap’

* Definer skærm
* Laver objekter dvs. kalder klasser (sender værdier for rgb, x,y-placering til klasse og tykkelse)
* Kalder funktioner fra klasse (funktion for indramning af menu, knapper og tykkelse)
* Funktion for mouseKlikket  
   if (mus er trykket på)  
   kald ’klikket’-funktion fra klasse for knap-objekterne

Klasse

* constructor  
   omdan værdier sendt fra set up til lokale værdier i klassen
* Funktion for knap  
   if(mus er inden for knap)  
   skalerer rgb-værdierne ned  
   else  
   brug rgb-værdierne sendt med fra set up  
    
   tegn rektangel  
    
   funktion for hvis der er klikket  
   if(mus er inden for knap)  
   sæt boolean-variable til true  
   return boolean-værdien  
    
   if(mus er inden for knap for forstørrelse af tykkelse)  
   kald funktion ’tykkere’  
    
   if(mus er inden for knap for mindskelse af tykkelse)  
   kald funktion ’tyndere’  
    
   if(boolean-variable er true)  
   skalerer rgb-værdierne ned  
   tegn ny rektangel over den gamle  
   funktion for hvis der er klikket  
   if(mus er inden for knap)  
   sæt boolean-variable til false  
   return boolean-værdien
* Funktion for linje  
   laver linje tykkelse  
   if(mus er før menu)  
   farv linje  
   lav linje
* Funktion for tykkelse på menuen  
   lav rektangel  
   skriv linje tykkelsen på rektanglet  
   skriv + og - på rektangelet to sider
* Funktion ’tykkere’  
   læg 0,5 til linjens tykkelse
* Funktion ’tyndere’  
   træk 0,5 fra linjens tykkelse
* Funktion ’tyndere’  
   træk 0,5 fra linjens tykkelse
* Funktion for indramning af menu  
   farv  
   lav rektangel

## 3.2.4 klasse beskrivelse og diagram

|  |
| --- |
| Button |
| -r: int -g: int -b: int -xPlus: int  -xPlus: int -strokeSize: int |
| #knap()  #klikket()  #vis(in strokeSize1 : int, in strokeSize2 : int)  #tykkelse()  #tykkere()  #tyndere()  #cubes() |

## 3.3 deling

## 3.3.1 github

For at vi kunne arbejde sammen brugte vi GitHub til at dele opdateringer af de ting, vi har ændret i koden. Dette hjalp os til at kunne arbejde separat og nemt dele det uden at bekymre os. Vi delte det med vores lærer, så det også er nemt for ham at få adgang til vores fremskridt.

A screenshot of a computer screen

Description automatically generated with medium confidence

# 4 test

# Konklusion

# 

# Bilag