

# Eksamen i Programmering 2025

## Projektbeskrivelse

**Emne:** Programmering

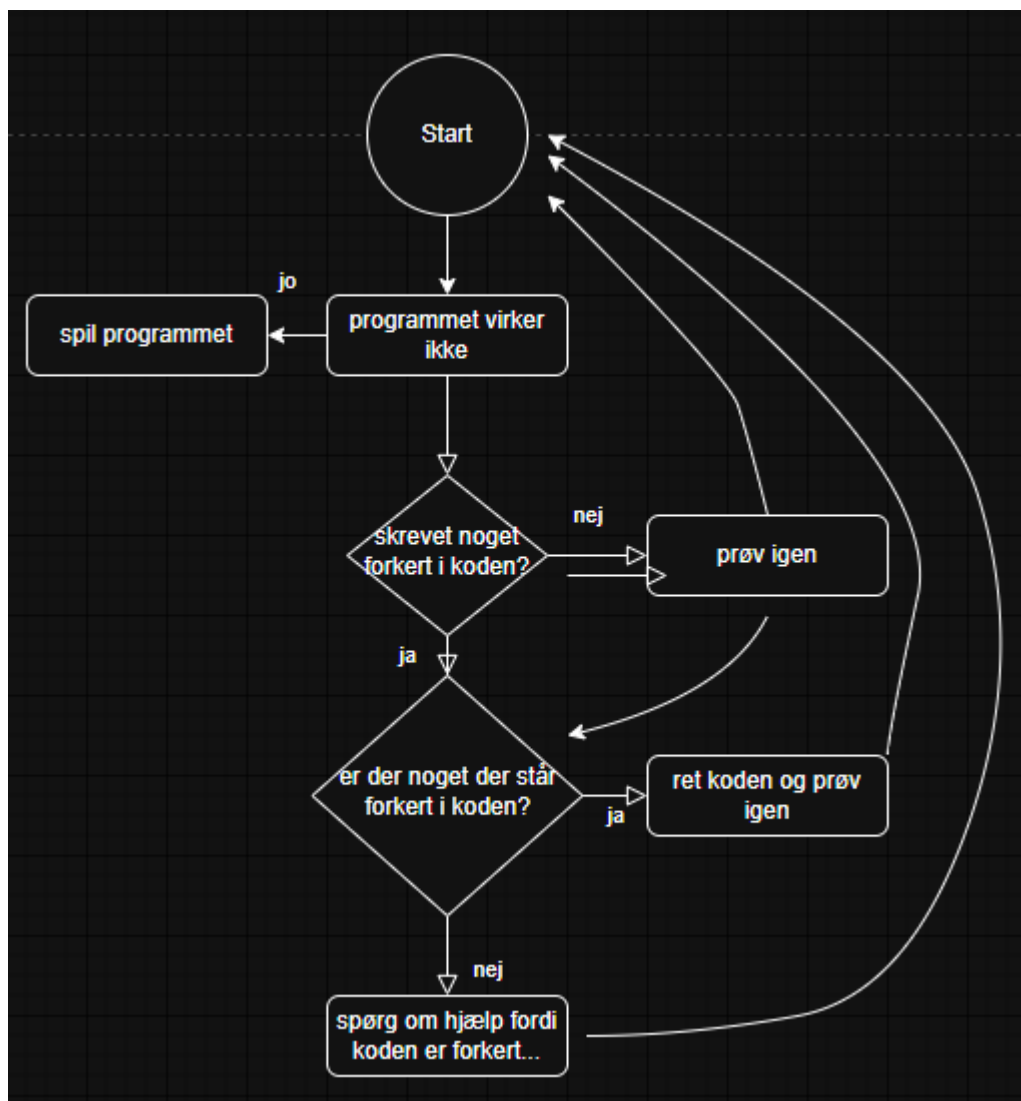
**Projektbeskrivelse:** Interaktiv Quiz om Programmering

**Emne:** Interaktiv læringsquiz om Programmering

**Beskrivelse:**

I det her projekt vil jeg udvikle en interaktiv quiz, der skal gøre læring om Programmering mere engagerende og tilgængelig for elever og studerende. Quizzen vil omfatte spørgsmål fra forskellige programmerings emner. Den interaktive quiz vil blive udviklet som en digital løsning, der kan tilgås via processing. Quizzen vil indeholde forskellige spørgsmålstyper såsom multiple choice, sandt/falsk og billedbaserede opgaver for at sikre variation og dybere forståelse. For at øge motivationen vil der være et pointsystem, hvor brugerne kan optjene point baseret på deres præstationer. Målgruppen for quizzen er primært gymnasieelever og studerende, men den kan også anvendes af andre, der ønsker at forbedre deres programmerings viden på en sjov og interaktiv måde. Projektet vil kræve research af relevante programmerings emner, design og udvikling af quizzen samt testning for at sikre en brugervenlig oplevelse. Ved at kombinere læring med interaktivitet håber jeg at gøre programmering mere spændende og tilgængeligt for flere mennesker

## Rutediagram



### Centrale funktioner og elementer

- **setup()** initialiserer canvas og definerer størrelsen på vinduet.
- **draw()** kaldes i en uendelig løkke og tegner den bevægelige firkant samt håndterer baggrunden og animationen.
- **mousePressed()** ændrer baggrundsfarven, når musen trykkes ned.
- **Brug af variabler** som x, direction, r, g, b til at styre position og farve.
- **if-sætninger** bruges til at vende bevægelsesretningen, når figuren rammer kanten.

## **Teknisk analyse**

Programmet demonstrerer forståelse i Processing, herunder hvordan man bruger variabler til at opdatere elementer. Brug af betingelser og musens input gør det muligt at skabe dynamiske og interaktive programmer. Det er desuden et godt eksempel på, hvordan små ændringer i kode kan skabe interaktive og sjove spil.

Farvekodningen viser forståelse for RGB-systemet i interaktionen med brugeren (via musetryk). Det viser kreativ brug af grundlæggende programmeringselementer til at skabe visuelle og funktionelle resultater.

## **Fejlkilder og forbedringsmuligheder**

En mulig forbedring kunne være at tilføje flere figurer eller elementer, der reagerer forskelligt på musens input fx en knap der går til et leaderboard, hvilket ville give programmet mere dybde og kompleksitet.

## **Konklusion**

Programmet opfylder kravene til en basal, men funktionel animation i Processing. Det demonstrerer de centrale kompetencer indenfor Programmering B på HTX, herunder variabelstyring, animation, brugerinteraktion og brug af grundlæggende strukturer som setup, draw og hændelsesstyring. Med mindre forbedringer kan det udvikles videre til et endnu mere avanceret interaktivt værk.

## **Krav specifikation**

### **Need to have**

Designs

Variation på spørgsmål

Forskellig sværhedsgrad

### **Nice to have**

Point

Lyd

Leaderboard

multiple choice

## Koden

Hoveddel

```
int questionnumber; // Hvilket spørgsmål brugeren er nået til
```

```
Answer[]Answr; // Array af svar-knapper (objekter)
```

```
Question[]questions; // Array med spørgsmål (og tilhørende svar)
```

```
int correctanswers; // Hvor mange rigtige svar brugeren har fået
```

```
int counter; // Bruges til at vise svaret i kort tid
```

```
boolean showsanswer; // Om svaret skal vises (true/false)
```

```
int selectedanswer; // Hvilket svar brugeren valgte
```

```
void setup() {
```

```
    size(800, 600); // Sætter vinduets størrelse
```

```
    textSize(20);
```

```
    fill(0);
```

```
    questions=new Question[] { // Initialiserer spørgsmål og svarmuligheder
```

```
        new Question("Hvad bruges setup() funktionen til i Processing?", "At tegne figurer",  
"At initialisere variabler og sætte skærmstørrelse", "At afslutte programmet", "At  
afspille lyd", 1),
```

```
        new Question("Hvad gør size(800, 600) i en sketch?", "Sætter farven på baggrunden",  
"Bestemmer tykkelsen på linjer", "Sætter vinduets bredde og højde til 800 x 600 pixels",  
"Zoomer ind på skærmen", 2),
```

```
        new Question("Hvad er forskellen på fill() og stroke() i Processing", "fill() farver  
baggrunden, stroke() slukker programmet", "fill() fjerner tekst, stroke() fjerner billeder",  
"fill() bruges til figurers inderside, stroke() til kanten", "De gør det samme", 2),
```

```
        new Question("Hvad gør background(255)", "Tegner en sort baggrund", "Sætter  
baggrunden til hvid", "Starter et loop", "Gør figurer gennemsigtige", 1),
```

```
        new Question("Hvordan tegner man en cirkel i Processing?", "drawCircle(100, 100,  
50)", "circle(100, 100, 50)", "ellipse(100, 100, 50, 50)", "oval(100, 100, 50)", 2),
```

```
        new Question("Hvad er koordinatsystemets udgangspunkt i Processing?", "Midten af  
skærmen", "Nederste højre hjørne", "Øverste venstre hjørne", "Øverste midtpunkt", 2),
```

new Question("Hvad gør line(0, 0, 100, 100)?", "Tegner en firkant", "Tegner en linje fra (0,0) til (100,100)", "Sletter en linje", "Tegner en cirkel", 1),

new Question("Hvilken datatype bruges til decimaler i Processing?", "int", "boolean", "char", "float", 3),

new Question("Hvad gør noLoop()? ", "Starter et nyt loop", "Stopper draw() fra at køre kontinuerligt ", "Stopper hele programmet", "Tegner en loopet form", 1),

new Question("Hvordan kan man vise tekst på skærmen i Processing?", "text(hej, 100, 100)", "write(hej, 100, 100)", "print(Hej)", "label(Hej, 100, 100)", 0),

new Question("Hvad bruges draw() funktionen til i Processing?", "At sætte skærmstørrelsen", "At initialisere variabler", "At tegne figurer én gang", "At opdatere og tegne indhold kontinuerligt", 3),

new Question("Hvordan laver man en cirkel, der følger musen?", "ellipse(mouseX, mouseY, 50, 50)", "circle(mouseY, mouseX, 50)", "mouse(ellipse, 50, 50)", "drawCircle(mouse.move, 50)", 0),

new Question("Hvad gør mousePressed() funktionen?", "Den tegner en figur ved klik", "Den ændrer skærmstørrelsen", " Den kaldes automatisk, når musen klikkes", "Den kører kun, når draw() stoppes", 2),

new Question("Hvad sker der, hvis man bruger frameRate(60)", "Programmet opdaterer 60 gange i minuttet", " Programmet crasher", "Skærmen vises i sort/hvid, når musen klikkes", " Programmet opdaterer 60 gange i sekundet", 3),

new Question("Hvordan laver man en farve med RGB i Processing?", "color(255, 0, 0)", "rgb(255, 0, 0)", "fillColor(255, 0, 0)", "colorRGB(255, 0, 0)", 0),

new Question("Hvordan gemmer man værdier i en array i Processing?", "int myArray = [1, 2, 3]", "array myArray = (1, 2, 3)", "int[] myArray = {1, 2, 3}", "newArray = 1,2,3", 2),

new Question("Hvordan laver man en funktion i Processing?", "create function()", "function myFunc() {}", "void myFunc() {}", "def myFunc()", 2),

new Question("Hvad bruges keyPressed() til?", "Til at trykke på musen", "Til at tegne med farver", "Til at fange tastaturinput", " Til at afslutte programmet", 2),

new Question("Hvordan tegner man en rektangel uden kant?", "noBorder()", "noStroke()", "noEdge()", "stroke(0)", 1),

new Question("Hvordan kan man få en figur til at bevæge sig over tid?", "Ved at placere den i setup()", "Ved at bruge delay()", "Ved at ændre dens position i draw()", "Ved at tegne den med print()", 2),

```

}; // Initialiserer fire svar-knapper

Answr=new Answer[]{
    new Answer(width/2-75/2-50, 180, 75, 25),
    new Answer(width/2-75/2+50, 180, 75, 25),
    new Answer(width/2-75/2-50, 280, 75, 25),
    new Answer(width/2-75/2+50, 280, 75, 25),
};
}

void draw() {
    background(0, 162, 232); // Blå baggrund

    if (questionnumber<questions.length) {
        textAlign(CENTER, CENTER);

        text( questions[questionnumber].getQuest(), width/2, 100); // Vis spørgsmålet

        displayAnswers(); // Vis svarmuligheder

        if (counter>0) { // Vis korrekt/forkert farve i kort tid

            counter-=1;

            if (counter==0) {

                questionnumber+=1;

                showsanswer=false;

            }

        }

    } else {

        background(0, 162, 255);

        textAlign(CENTER, CENTER);

        text( "quiz slut", width/2, height/2); // Quiz er slut

        text( "antal rigtige"+correctanswers+"/"+questions.length, width/2, height/2+50); //vis
        resultat

    }
}

```

```

}

//funktion

void displayAnswers() {
    //String[] answers=question.getAnswers();

    int i=0;

    for (Answer answer : Answr) {

        fill(255); // Hvid baggrund for svar kasserne

        if (answer.isPointInside(mouseX, mouseY)) {

            fill(227); // Lysere baggrund, hvis musen er over

        }

        if (showsanswer==true) { // om svaret skal vises

            if (selectedanswer==i) {

                if (selectedanswer==questions [questionnumber]. getCorrectAnswer()) {

                    fill (0, 155, 0); // Grønt – korrekt

                } else {

                    fill(255, 0, 0); // Rødt – forkert

                }

            }

        }

        rect(answer.x, answer.y, answer.w, answer.h);

        fill(0);

        text(questions[questionnumber].getAnswers()[i], answer.x+answer.w/2,
        answer.y+answer.h/2);

        i++;

    }

}

void mousePressed() {

    for (int i = 0; i < Answr.length; i++) {

```

```

if (Answr[i].isPointInside(mouseX, mouseY)) {
    println(i); // Udskriv hvilket svar brugeren kikkede på
    if (questionnumber+1<questions.length+1) {
        counter=30; // Start en lille "pause" på 30 frames så folk kan se om det er forkert
        eller rigtig
        selectedanswer=i;
        showsanswer=true;
        if (questions[questionnumber].getCorrectAnswer()==i) {
            correctanswers+=1; // Hvis korrekt, giv point
        }
    }
}
}
}
}
}
}
}

```

2 klasse

class Question { //definerer en ny klasse i Processing (Java), kaldet Question

//attributer

String quest= "hvad er en byte";

String[] answers = new String [4];

int correctAnswer=2;

//konstuktør

Question(String question, String answer1, String answer2, String answer3, String  
answer4, int correctanswer) {

answers[0] = answer1;

answers[1] = answer2;

answers[2] = answer3;

answers[3] = answer4;

correctAnswer=correctanswer;

quest=question;



```

}

//metoder

String getQuest() {
    return quest; // Returnerer spørgsmålet (tekst)
}

String[] getAnswers () {
    return answers; //Returnerer alle fire svar i et array
}

int getCorrectAnswer() {
    return correctAnswer; //Returnerer index-værdien for det korrekte svar (0-3)
}
}

```

3 klasse

class Answer { //definerer en ny klasse, som bruges til at repræsentere en klikbar knap

//fire variabler beskriver en rektangel

int x;

int y;

int w;

int h;

Answer(int x, int y, int w, int h) {

this.x = x;

this.y = y;

this.w = w;

this.h = h;

}

boolean isPointInside(int px, int py) { //tjekker om musens position er inde i knappen

return (px > x && px < x+w) && (py > y && py < y+h);

```
//px = musens x-position  
//py = musens y-position  
}  
}
```