

Prog-C Synopsis



HITX

TEKNISK GYMNASIUM • HJØRRING

Elev:

Elias Deth Melchiorson

Gruppemedlemmer:

Dan Juul Christensen

Simon Beith Agerlyst

Underviser:

Nanna Godiksen Dahl

Underskrift:

Indholdsfortegnelse

Resume:	3
Problemformulering:	4
Projektafgrænsning:	4
Overvejelser:	4
Tidsplan:	5
Gruppemedlemmers hovedfokus:	5
Kravspecifikation:	5
Funktionsbeskrivelse	6
Dokumentation:	6
Test af programmet:	7
Konklusion:	8
Kildeliste:	8
Bilag:	9
Flowchart:	9
Tilstandsdiagram:	10
Usecases:	10

Resume:

Dette projekt drejer sig om udviklingen af et programmeret Othello-spil, der har til formål at bekæmpe kedsomhed ved at tilbyde en udfordrende og engagerende spiloplevelse. Othello er et strategisk brætspil, hvor spillerne kæmper om at tage kontrol over et bræt ved at placere brikker og vende modstanderens brikker til deres egen farve.

Spillet's succes afhænger af flere faktorer, herunder spillets kompleksitet og sværhedsgrad, brugervenlighed og grafisk design. Derfor vil projektet involvere udvikling af et effektivt spil Interface og en brugervenlig grænseflade, der vil motivere brugeren til at spille spillet flere gange.

Det endelige mål for projektet er at udvikle et Othello-spil, der kan modstå kedsomhed og give brugerne en underholdende spiloplevelse.

Problemformulering:

Hvordan kan man udvikle et programmeret Othello-spil, der kan bekæmpe kedsomhed ved at tilbyde en udfordrende og engagerende spiloplevelse til brugerne?

Projektafgrænsning:

Dette projekt vil fokusere på udviklingen af et programmeret Othello-spil, der kan modstå kedsomhed og give brugerne en underholdende spiloplevelse. Projektet vil primært involvere udvikling af et effektivt spil Interface og en brugervenlig grænseflade, der vil motivere brugeren til at spille spillet flere gange.

Projektet vil ikke omfatte udviklingen af avancerede funktioner, såsom multiplayer-tilstand eller integration med sociale medieplatforme. Det vil heller ikke omfatte grafisk design af høj kvalitet, men vil fokusere på en enkel og overskuelig interface, der fungerer effektivt.

Projektet vil også have en tidsmæssig afgrænsning, og der vil være begrænsninger for antallet af spil niveauer og spilfunktioner, der kan implementeres inden for den fastsatte tidsramme.

Endelig vil projektet fokusere på udvikling af Othello-spillet på Desktop.

Overvejelser:

1. Målgruppe: Det er vigtigt at overveje, hvem spillet er rettet mod, da dette vil påvirke spillets kompleksitet, sværhedsgrad og design. Hvis spillet er rettet mod børn, vil man sandsynligvis ønske at gøre det mere visuelt tiltalende og muligvis reducere spillets kompleksitet.
2. Platform: Othello-spil kan udvikles til forskellige platforme, herunder desktop, mobil og online. Valg af platform vil påvirke spillets funktioner og design, da der kan være begrænsninger i forhold til skærmstørrelse, tekniske krav og brugerinteraktion.
3. Spil Interface: Udviklingen af et effektivt spil interface er afgørende for spillets succes. Spil Interfacen skal sikre, at spillet er udfordrende og engagerende, men stadig muligt at spille og vinde. Dette vil skabe en god balance mellem spillets sværhedsgrad og brugervenlighed.
4. Interface: En brugervenlig grænseflade vil være afgørende for spillets succes. Grænsefladen skal være let at navigere og intuitiv at bruge, og samtidig skal den også have et tiltrækkende design, der vil tiltrække brugerne og motivere dem til at spille spillet igen.
5. Tids- og ressourcestyring: Det er vigtigt at planlægge og administrere projektet effektivt, herunder at fastlægge en realistisk tidsplan. Der skal også tages hensyn til eventuelle tekniske begrænsninger og tilgængelige ressourcer, som kan påvirke projektets succes og fremgang.

Tidsplan:

Projektbeskrivelse	9		6								
Use Cases				4,5	3						
Diagrammer				4,5	3						
Pseudokode					3			3			
pænt program								2		2	2
Kommentare								2		2	2
synopse								6		6	6
test program										2	2
uger	uge 9	uge 10	uge 11	uge 12	uge 13	uge 14	uge 15	uge 16	uge 17	uge 18	uge 19

Gruppemedlemmers hovedfokus:

Dan: Dan har haft sit fokus på programmering af spillet, han har skrevet pseudokode og fremstillet koden. Dan har samt også hjulpet med at fremstille og budt på nogle af diagrammerne såsom tidsplanen og flowchartet

Elias: Jeg og Simon har haft vores fokus på det skriftlige og det med at fremstille spørgeskemaer, diagrammer og dokumentation og givet lidt input til hvordan programmet kan se ud og hvilke krav det skal være

Simon: Simon og jeg har haft vores fokus på det skriftlige og det med at fremstille spørgeskemaer, diagrammer og dokumentation og givet lidt input til hvordan programmet kan se ud og hvilke krav det skal være

Kravspecifikation:

1. Det skal være muligt at vinde, når hele pladen er fyldt. Dette vil blive testet ved at fylde hele spillepladen ud med de forskellige brikker.
2. Spillet skal kunne fortsætte, indtil hele pladen er fyldt. Dette vil blive testet ved at trykke på de tomme felter, indtil hele pladen er fyldt op.
3. Man skal kunne spille to sammen. Dette vil blive testet ved at spille et spil mellem to personer.
4. Der skal være to hvide og to sorte brikker i de fire kasser i centrum. Dette vil blive testet ved at starte spillet og se, om de er til stede.
5. Det skal ikke være muligt at placere en brik på et felt, hvor der allerede er en brik. Dette vil blive testet ved at trykke på et felt, hvor der allerede er en brik til stede.

Funktionsbeskrivelse

1. Lav systemKrav:
 - Spillet skal være let at køre på et bredt udvalg af computere uden at computeren skal have en hurtig eller avanceret hardwarekonfiguration.
 - Det skal ikke kræve en internetforbindelse eller Wi-Fi Forbindelse for at kunne spilles, hvilket gør det let tilgængeligt uanset hvor man er.
2. Simple regler og design:
 - Spillet skal have enkle og letforståelige regler, så det kan spilles af et bredt omfang af mennesker som ikke har en tidligere spille-erfaring.
 - Designet skal være tiltalende og have et visuelt behageligt udseende uden at være alt for komplekst eller for overvældende.
3. Tankegang og planlægning:
 - Spillet skal give en udfordring til spillernes tankegang, planlægningssevner og problemløsning. Spillet skulle derved styrke deres kognitive evner.

4. Lokal multiplayer-funktion:

- Spillet skal understøtte lokal multiplayer, så det skulle derfor kunne spilles af to personer, der sidder ved den samme computer.

5. Underholdningsværdi:

- Spillet skal kunne underholde spillerne og give dem lyst til at spille mere og derved formindske spillernes kedsomhed

Dokumentation:

Algoritmiske forhindringer:

Under fremstilling af programmet var det første problem at spillet 'crashede', vi undersøgte at programmet stoppede når man prøvede at placere en brik uden for selve spillepladen (Hertil havde vi en mørk border omkring pladen). Jeg foreslog at vi lavede så pladen fyldte hele programvinduet, som virkede.

Det mest komplicerede var at få brikkerne til at ændre til den rigtige farve, når de skulle vendes. Når en række brikker blev flankeret ændrede deres farve, men det gjorde de brikker der skulle forblive deres originale farve også. Og at programmere så brikkerne kunne blive flankeret var også et problem som skulle blive løst.

Programoversigt:

Programmet er en implementering af Othello, et populært strategisk brætspil, hvor to spillere konkurrerer om at få flest brikker på brættet ved at placere deres egne brikker og vende modstanderens brikker. Programmet er udviklet i Processing 4 og er designet til at bekæmpe kedsomhed ved at tilbyde brugeren en sjov og udfordrende spiloplevelse.

Funktionalitet:

Programmet giver brugeren mulighed for at spille Othello. Programmet giver mulighed for at nulstille spillet og starte en ny runde. Når brugeren åbner programmet, starter spillet og man kan gå i gang med at placere brikker med det samme. Programmet er designet til at give brugeren en optimal spiloplevelse, som inkluderer en række muligheder og funktioner, som gør spillet både sjovt og udfordrende. Spillet slutter, når brættet er fyldt. Når spillet slutter, viser programmet vinderen og antallet af brikker, som hver spiller har på brættet.

Interface:

Interfacen er enkel og let at navigere, med en brugervenlig menu og interaktive knapper, der gør det nemt for brugeren at starte og spille og nulstille spillet. Brugeren kan også se antallet af brikker for hver spiller og følge med i spillets status på brættet. Når spillet starter, vises brættet på skærmen, og brugeren kan placere deres brikker ved at klikke på det ønskede felt på brættet. Når brikkerne flankeres ændres farven på brikken og brikken tælles med.

Algoritme:

Algoritmen er udviklet til at følge de grundlæggende regler for Othello og tilbyde en udfordrende spiloplevelse for brugeren. Algoritmen er designet til at træffe intelligente beslutninger baseret på brættets nuværende tilstand og brugerens træk, og tilbyder tre

Elias D.M 2.x
HTX Hjørring EUC
Programmering C

forskellige sværhedsgrader, der varierer i forhold til computerens evne til at træffe disse beslutninger.

Test:

Programmet er testet grundigt for at sikre, at det fungerer korrekt og leverer den ønskede spiloplevelse. algoritmen er blevet testet for at sikre, at den er i stand til at identificere gyldige og ugyldige træk og tilbyde en udfordrende spil oplevelse for brugeren. interfacen er også blevet testet for at sikre, at den er let at navigere og intuitiv at bruge.

Konklusion

Samlet set er programmet en effektiv implementering af Othello, der leverer en sjov og udfordrende spiloplevelse til brugeren. Med en intuitiv brugergrænseflade og en intelligent algoritme er programmet i stand til at bekæmpe kedsomhed og underholde brugeren. Programmet er udviklet i Processing 4 og kan tilpasses til at inkludere flere funktioner, såsom Online Multiplayer.

Test af programmet:

1. Spil interface test.

Test af spillets Interface for at sikre, at spillet fungerer korrekt og er i stand til at identificere gyldige og ugyldige træk. Dette kan omfatte at teste forskellige træk sekvenser og sikre, at spillet opfører sig som forventet.

2. Fejlhåndtering Test:

Test af spillets fejlhåndtering for at sikre, at det kan håndtere uventede situationer, såsom input fejl eller tekniske problemer. Dette kan omfatte at teste forskellige scenarier, såsom mistet internetforbindelse eller systemnedbrud, for at sikre, at spillet fortsætter med at fungere korrekt.

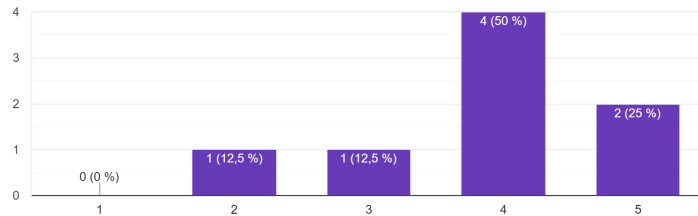
1. Brugertests:

Endelig vil det være vigtigt at udføre brugertests for at evaluere, om spillet giver den ønskede spiloplevelse og leverer den forventede underholdning. Brugertests kan omfatte spørgeskemaundersøgelser eller interviews med brugerne for at evaluere deres tilfredshed med spillet og identificere eventuelle områder til forbedring.

Vi lavede et spørgeskema, som folk ville udfylde efter at have testet programmet.

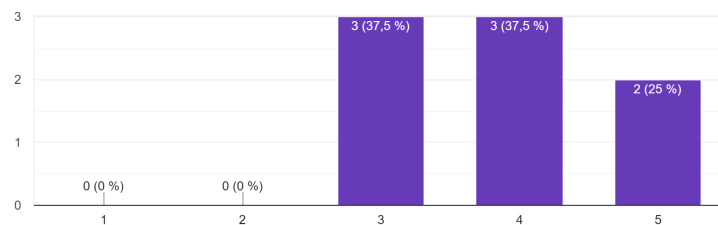
1. Spørgsmål: Udseende. 1-5. 5 Højest 1 mindst

Udseende
8 svar



2. Spørgsmål: Fornøjelse 1-5. 5 Højest 1 mindst

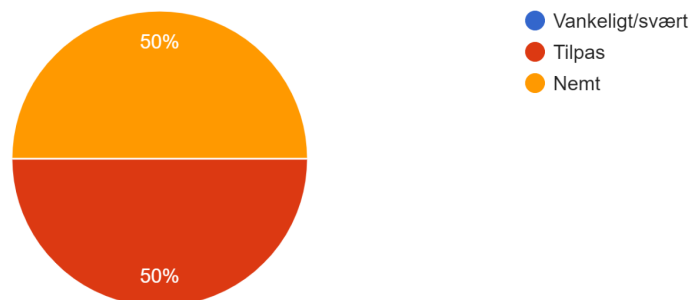
Fornøjelse
8 svar



3. Spørgsmål: Håndtering

Håndtering af spillet (Er det nemt at bruge og udnytte?)

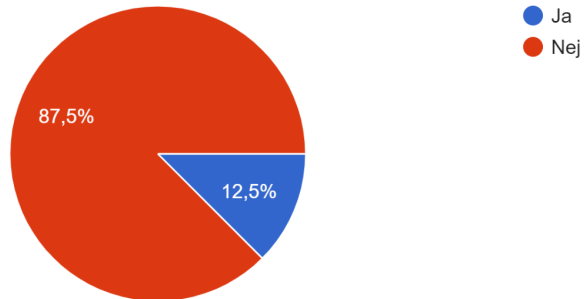
8 svar



4. Spørgsmål: "Har du spillet Othello før?"

Har du spillet Othello før?

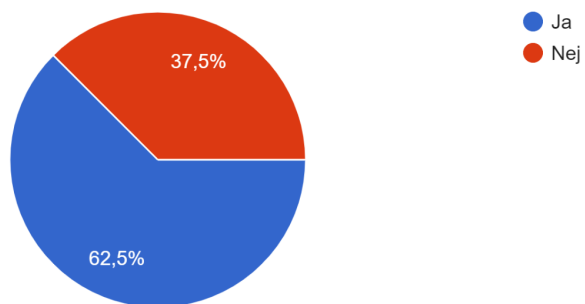
8 svar



5. Spørgsmål: "Vil du spille det igen?"

Vil du spille det igen

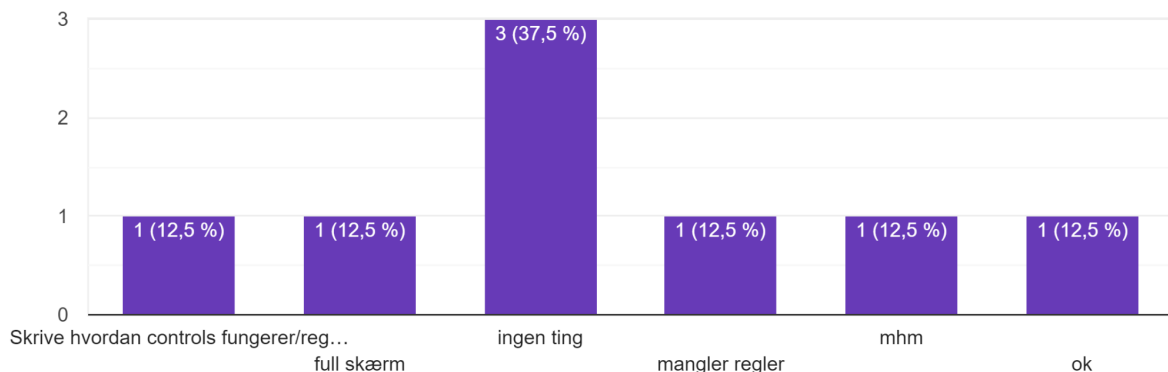
8 svar



6. Spørgsmål: (Kort svar fra bruger)

Hvad kunne forbedres?

8 svar



Konklusion:

Udviklingen af et Othello-spil, der kan bekæmpe kedsomhed, kan være en interessant og udfordrende opgave. Et effektivt spil Interface og Interface er afgørende for at opnå den ønskede spiloplevelse, og det er vigtigt at overveje målgruppen og platformen for at skabe en tilpasset spiloplevelse.

Tids- og ressourcestyring er også afgørende for at sikre, at projektet forbliver på sporet og leverer et færdigt produkt til tiden og indenfor budgettet.

Når spillet er færdigt, vil det være vigtigt at teste det grundigt for at sikre, at det fungerer korrekt og leverer den ønskede spiloplevelse. Dette vil kræve test af spil, Interfacen, Interfacen, sværhedsgrader, fejlhåndtering og brugertests.

Samlet set kan udviklingen af et Othello-spil være en spændende og givende opgave, der kan bidrage til at bekæmpe kedsomhed og underholde brugerne. Ved at følge en effektiv udviklingsproces og teststrategi kan man opnå succes med spillet og skabe en tilfredsstillende spiloplevelse for brugerne.

Kildeliste:

Processing 4 Reference

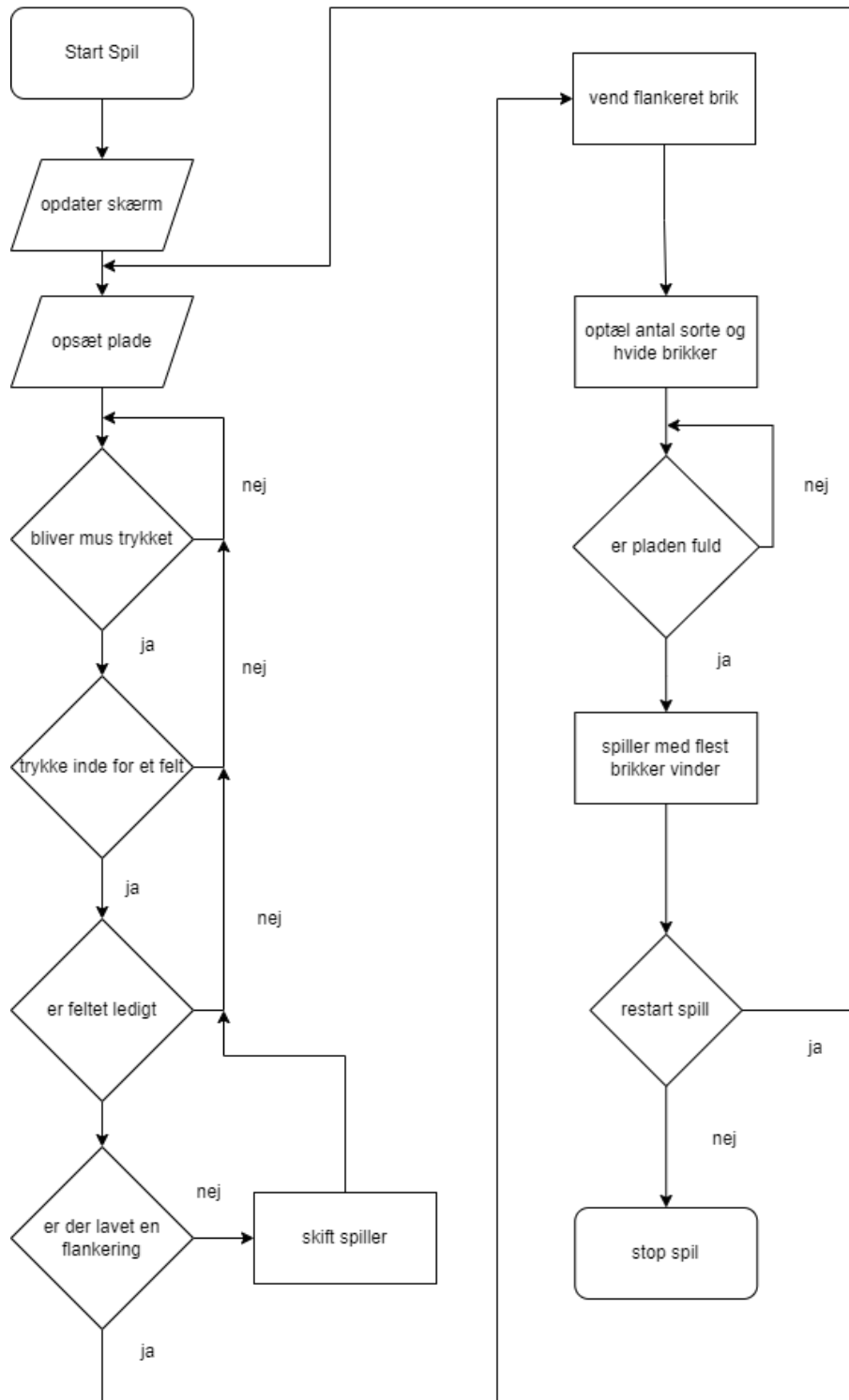
<https://processing.org/reference>

Java Keyword

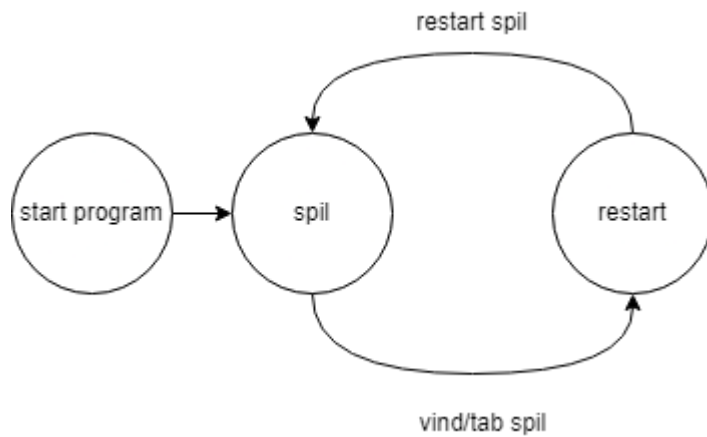
https://www.w3schools.com/java/ref_keyword_this.asp

Bilag:

Flowchart: Vores flowchart giver en visuel repræsentation kort om hvordan programmet virker, og hvordan det er programmeret.



Tilstandsdiagram:



UseCases: UseCases viser hvad brugeren (Actor) gør, som computeren gør samt brugerens handlinger.

