



## **Memory Card Game Project**

***STUDIERICHTING ELEKTROTECHNIEK***

Samenstellers: Sokromo Cynthiëne 20230010  
Thakoerdatt Shrija 20230110  
Jacott Jamiro 20220203

Vak: Object Georiënteerd Programmeren 1  
Docent: Winston Soetosenojo Msc.  
Datum: 06-08-2024

*Paramaribo, Suriname*

## **Idee achter de Memory Card Game:**

Als groep wilden wij een game creëren dat eenvoudig is en niet uit ingewikkelde functies bestaat, verder wilden wij een game die leuk te spelen is en die aantrekkelijk is. Verder wilden wij ook iets ontwikkelen dat de gebruiker een uitdaging geeft. Om een game te krijgen dat aan al deze eisen voldoet hebben wij een aantal games onderzocht. Uiteindelijk hebben we besloten om een eenvoudige memory card game te programmeren.

## **Waarom dit spel gekozen?**

We hebben bewust dit soort games programmeren omdat wij iets eenvoudig wilden creëren. Als studenten hebben wij niet veel tijd door allerlei vakken, een eenvoudige Memory card game bestaat dus niet veel tijd en voldoet aan al onze eisen

**Naam van de game :** Card Match Game (MEMORY)

## **Project Doelen:**

Het doel van dit project is het ontwikkelen van een Card Match Game in C++ met behulp van SFML. Het doel van dit spel is om spelers te testen door kaarten te matchen in paren.

## **Specifieke Doelen:**

- \* Een visueel aantrekkelijk kaartspel maken waarbij spelers kaarten moeten matchen door ze om te draaien.
- \* SFML implementeren voor grafische weergave.
- \* Het spel moet de mogelijkheid hebben om kaarten om te draaien, te vergelijken, en het spel moet eindigen als alle paren gevonden zijn.

## **Spelregels:**

1. Het spel begint met alle kaarten omgekeerd, zodat de speler de afbeeldingen niet kan zien.
2. De speler kan twee kaarten tegelijk draaien door op ze te klikken.
3. Nadat twee kaarten zijn omgedraaid, worden ze vergeleken om te controleren of ze een paar vormen.
4. Als de twee omgedraaide kaarten een paar vormen (identieke afbeeldingen), blijven ze omgedraaid en worden ze uit het spel verwijderd en krijgt de speler 1 punt
5. Als de kaarten geen paar vormen, worden ze weer omgekeerd en is het de andere spelers beurt.
6. Het spel eindigt wanneer alle kaarten in paren zijn gematched.
7. De winnaar is de speler met de meeste punten.

## Overzicht van de Memory Card game code:

De code betreft een eenvoudig memory card game die twee spelers om de beurt kaarten laat omdraaien om paren te vinden. Het spel is gebouwd met behulp van de SFML (Simple and Fast Multimedia Library) in C++. Voor dit project hebben wij de Visual Studio Code (VS Code) Community 2022 open source editor gebruikt.

Onze programma bestaat uit 3 classes o.a:

1. **Player Class:** Beheert spelersnamen en scores. Methodes omvatten het opvragen van naam en score, het verhogen van de score en het wisselen van spelers.
2. **Card Class:** Beheert de eigenschappen van een kaart, zoals positie, vorm en of de kaart is omgedraaid. Methodes omvatten het positioneren, tekenen, omdraaien van de kaart en het instellen van texturen.
3. **MemoryGame Class:** Bevat de Hoofd Logica van het spel. De 'Run' functie start het de hoofdgame loop. De 'draw' functie is voor het tekenen van de kaarten, scores en huidige spelerstatus. De 'handleEvents' functie zorgt ervoor dat de gebruikersinvoer zoals muisklikken verwerkt worden. En de 'checkMatch' functie vormt een controle, deze functie controleert of de twee omgedraaide kaarten een paar vormen en verwerkt de uitkomst.

Naast de 3 classes bevat onze programma ook een welkomscherm code. Deze code en de player class zijn door middel van header files in onze main programma geïmplementeerd.

De welkomscherm code zorgt voor een aantrekkelijk welkomstscherm met een voortgangsbalk, die de gebruiker informeert dat het spel wordt geladen. Het gebruikt eenvoudige console-opdrachten en Windows-specifieke functies om de visuele presentatie te verbeteren. De 'welcome\_part.h' creëert een welkomstscherm voor een memory card game in de console. Het stelt de console in met 'setup Console()', toont een welkomstbericht en spelersnamen met 'displayWelcomeScreen()', en animeert een voortgangsbalk met 'displayProgressBar()'. Deze functies verbeteren de visuele presentatie en gebruikerservaring van het spel.

## Spelverloop:

1. **Initialisatie:** Texturen voor de voorkant en achterkant van de kaarten worden geladen. De kaarten worden aangemaakt en willekeurig gepositioneerd.
2. **Spel Loop:** De 'run' methode zorgt ervoor dat het spel blijft draaien zolang het venster open is. Gebruikersinvoer wordt verwerkt, de weergave bijgewerkt en gecontroleerd of er een match is.
3. **Gebeurtenis Verwerking:** Bij een klik op een kaart wordt gecontroleerd of deze nog niet is omgedraaid. De kaart wordt omgedraaid en de index opgeslagen. Na twee kaarten begint een timer om de kaarten te vergelijken.

4. **Match Controle:** Na de timer worden de kaarten vergeleken. Bij een match wordt de score verhoogd; anders worden de kaarten teruggedraaid en wisselt de beurt.
5. **Weergave Bijwerken:** De 'draw' methode tekent alle kaarten, scores en de huidige spelerstatus.
6. **Resetten van het Spel:** De 'resetGame' methode het schudt de kaarten en zet de scores en spelers terug naar de beginstatus.

### **Process van het project:**

De groepscoördinatie tijdens dit project verliep helaas niet soepel. Het was voor ons erg moeilijk om samen te komen en gezamenlijk aan het project te werken. Bovendien verliep de communicatie binnen de groep niet goed. Desondanks hebben we het project uiteindelijk toch weten af te ronden, hoewel het een uitdagend proces was.

### **Participatie van de Groepsleden:**

**Jamiro:** Player class gemaakt en deels gewerkt aan de document

**Shrija:** Memorygame class, beide headers gemaakt en Document geschreven

**Cynthiëne:** Card class gemaakt, gewerkt aan de Github en Powerpoint gemaakt.

### **Ervaring tijdens het project:**

Het ontwikkelen van de memory card game was een leerzame ervaring. Hoewel het project enkele uitdagingen met zich meebracht, zoals problemen met groepscoördinatie en communicatie, vond ik het overall een leuke en waardevolle ervaring. Het was interessant om te werken met SFML en C++ om een functionele en visueel aantrekkelijke game te creëren. Ondanks de tijdsdruk en de complexiteit van sommige taken, genoten we van de creatie en implementatie van een spel dat zowel eenvoudig als uitdagend is.

### **Github:**

<https://github.com/CynthieneSokromo/Groep-C.-OGP1>