

Kwaliteit

Gebrek aan kwaliteitsvolle code betekent **minder punten**.

De code moet gemakkelijk te lezen en te onderhouden zijn en voldoet aan de programmeerregels.

Het project moet builden zonder fouten en zonder waarschuwingen.

Er mag geen code aanwezig zijn die niets te maken heeft met deze opdracht (geen kopie / plakken van andere oefeningen).

Namen van variabelen, functies, enumerations, structs,.. moeten **zinnig** zijn en de overeengekomen **naamgevingsregels** volgen.

Zorg ervoor dat er **geen geheugenlekken** zijn, alle **dynamische gealloceerde objecten** moeten worden **verwijderd** voordat het programma wordt afgesloten.

Het is niet toegestaan om C++ functies te gebruiken die nog niet in de lessen zijn besproken. Bijvoorbeeld: std::vector, smart pointers...

Online of Onsite

Indien je dit examen online aflegt, ben je verplicht aanwezig op het afgesproken online platform (discord <https://discord.gg/HUSaYkXSQD>). Je zal een webcam en microfoon nodig hebben om je te identificeren.

Technische voorschriften

Het is belangrijk dat u de opdracht uitvoert zoals gevraagd, zodat we kijken of u bepaalde programmeervaardigheden hebt verworven. Als uw oplossing werkt, maar als u het op een andere manier hebt opgelost, krijgt u minder punten. Dit omvat bijvoorbeeld: het creëren en gebruiken van de beschreven arrays, classes enumerations, ...

Gegeven

Een zip-bestand met een werkend voorbeeld exe, resources en bronbestand(en).

Het project maken

1. Maak een visual studio project met het framework, met behulp van de bestanden in 00_General/FrameworkFunctions_V0.2.4.zip. De naam van het project is **Prog1DAExx_Exyy_Familienaam_Voor naam**, ter vervanging van **xx** met uw groepsnummer, **Familienaam** door uw familienaam, **voornaam** door uw voornaam, **yy** door de examen nummer zoals het kan worden gevonden in de kop van dit document.
2. Plaats uw **volledige naam** en **groep** in de titel van het **venster**.

Assignment

Samenvatting:

Dit is een spel waarbij een kip een drukke straat probeert over te steken.

1. Je naam

Voeg in Game.h je naam en groep toe aan de titelbalk van het venster.

2. Teken de straat achtergrond

Laad het plaatje street.png en teken het op positie 0,0

3. De tekst

Alle tekst die op het venster getoond moet worden heeft een grootte van 40 en gebruikt het bijgeleverde consola font.

4. State van het spel

Voeg een enumeration toe die de toestand van het spel weergeeft. Het heeft de toestanden: spelen, pauzeren, winnen, verliezen, reset. Reset is de beginwaarde van de variabele van de enumeratie.

5. De kip

De kip moet worden geïmplementeerd in de Game bestanden. Het is geen klasse.

Kip eigenschappen:

- Afbeelding is **chicken.bmp**.

- De initiële x positie is de helft van de breedte van het venster. De initiële y-positie is `m_WindowHeight - 20`.

- De snelheid is 100 pixels per seconde in elke richting.

Laad de afbeelding Chicken.bmp. Het heeft drie frames. Gebruik voorlopig alleen het eerste, meest linkse frame. Pas aan het eind van deze opdracht vragen we je om ook de andere frames te gebruiken, omdat die taak een lagere prioriteit heeft. Teken de kip zo dat zijn standaard (voorste) frame op de stoep verschijnt, bovenaan in het venster.

Wanneer de pijltjestoetsen links of rechts worden ingedrukt, beweegt de kip in deze richtingen.

Als de pijltjestoets omlaag wordt ingedrukt, beweegt de kip naar beneden en steekt de straat over.

Door gelijktijdig op omlaag en links of rechts te drukken, wordt de beweging gecombineerd.

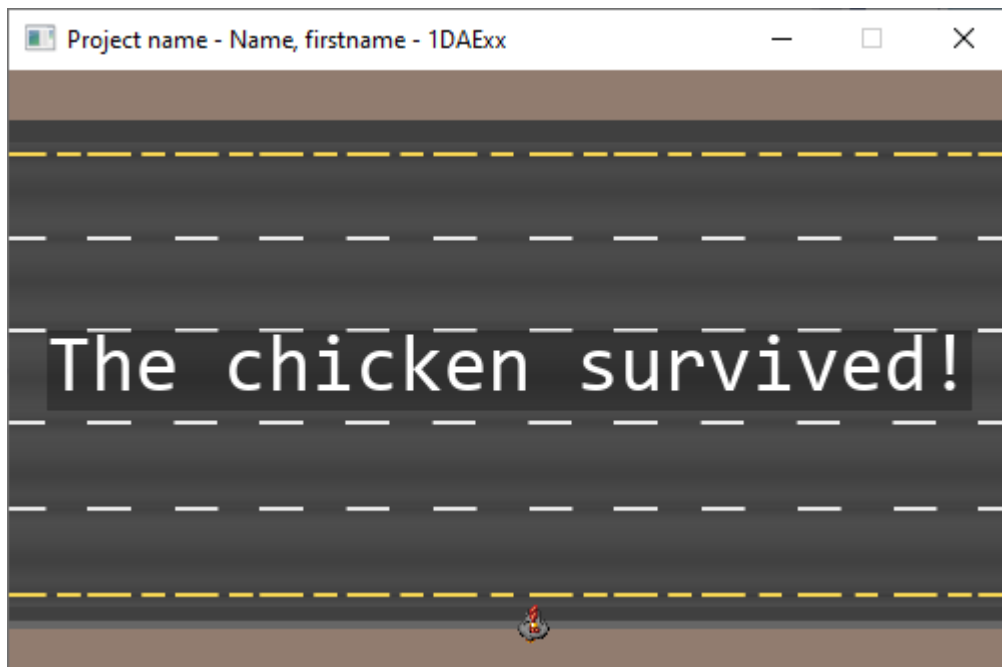
De hoogte van het onderste trottoir is 43 pixels. Als de kip het trottoir onderaan de straat bereikt,

- kan ze niet meer naar beneden, links of rechts bewegen,

- wordt een tekst getekend "De kip heeft het overleefd!", gecentreerd op het raam, op een verduisterde rechthoek om de leesbaarheid te vergroten.

Door op een willekeurig moment op de 'r'-toets te drukken, wordt de kip weer teruggezet naar zijn start positie op de bovenste stoep en kan hij weer bewegen. Als er tekst werd getoond, verdwijnt deze. De kip kan dan de straat weer oversteken.

Druk de toets binding en speelinstructions af in de console.



6. De voertuig-klasse:

Je kunt verder gaan zonder een klasse te maken, maar dan verlies je punten. Maak een Vehicle class met behulp van de gegeven Vehicle cpp en h bestanden. De public interface mag niet gewijzigd worden. Dat is het publieke deel van de definitie van de klasse. Je kunt private member functies en private member variabelen toevoegen.

6.1 Constructor:

Naast de hoofdtaken van een constructor, construeer je de image path/name string uit het id argument. Laad de textuur met behulp van die string.

Het voertuig is niet actief; het beweegt niet wanneer het spel start.

De m_Speed is een willekeurige waarde in het inclusieve interval [m_MinSpeed, m_MaxSpeed], waarbij m_MinSpeed 20 is en m_MaxSpeed 200.

De initiële x-positie is 0.

6.2. Draw member function:

Teken de textuur op de positie gedefinieerd door m_Position.

6.3 SetYPos member function:

Deze functie werkt de m_Position.y waarde bij.

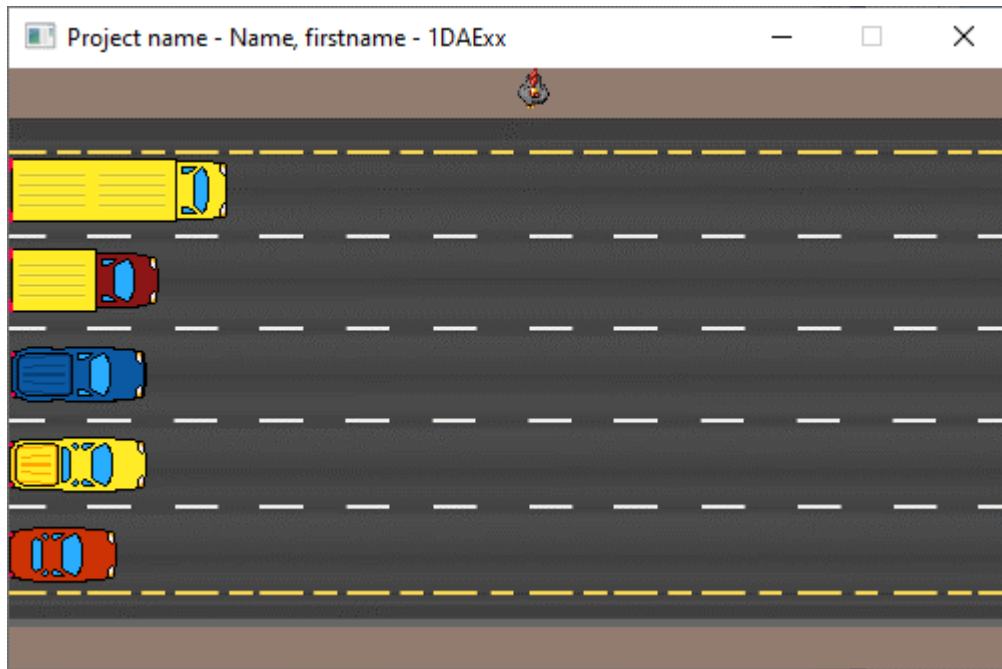
7. Voertuig objecten toevoegen aan het spel

Voeg een array van 5 voertuig pointers toe in de Game bestanden en maak 5 voertuig objecten op de heap terwijl je elke pointer in de array opslaat. Je kunt ervoor kiezen om geen array te gebruiken, maar dan verlies je punten. Elk voertuig-object moet een andere afbeelding laden door een nummer door te geven in zijn constructor. Voor nu, bij het creëren van het eerste voertuig geeft 1 door als parameter tot de constructor, creatie van het tweede voertuig 2, enz...

Update elke voertuig y-positie om het voertuig gecentreerd op een rijstrook te plaatsen. Het wegdek meet 43 pixels en elke rijstrook meet 45 pixels. Elk voertuig staat op een andere rijstrook. Het eerste voertuig staat op de laagste rijstrook, en elk volgend voertuig staat op de rijstrook daarboven, zoals je kunt zien op de schermafbeelding hieronder. Gebruik een lus hiervoor.

8. Teken de voertuigen.

Ze moeten onder elkaar staan. Zo zou het venster eruit moeten zien als je de applicatie uitvoert:



9. Implementeer de member functie Reset.

In deze functie:

- De spawn x-positie wordt willekeurig gekozen links van het zichtbare venster van 0 tot -300.
- De nieuwe snelheid wordt willekeurig gekozen tussen 20 en 200 pixels per seconde.
- Het voertuig wordt actief.

10. Implementeer de member functie Vehicle Update

Voer de nodige stappen uit zodat het voertuig naar rechts beweegt als het actief is. Wanneer het voertuig het venster verlaat, schuift het terug vanaf de linker rand van het venster.

11. In Game.cpp: de r-toets

Wanneer de r-toets wordt ingedrukt, wordt de reset functie gecalled en moeten ze opnieuw van links naar rechts bewegen.

12. InGame.cpp: Botsingen

Voeg de nodige stappen toe aan de Game Update functie om botsingsdetectie te hebben tussen de textuur van de voertuigen en de middenpositie van de kip.

Wanneer er een botsing is:

- stopt het botsende voertuig
- kan de kip niet langer bewegen
- negeren de andere voertuigen de botsing en blijven bewegen
- verschijnt er een witte tekst op een transparante zwarte achtergrond "De kip is gestorven".

13. Einde van het spel

Als de kip met succes de straat is overgestoken, verschijnt er een tekst op een doorzichtige zwarte achtergrond: "De kip heeft het overleefd". Alle voertuigen blijven rijden.

14. De r-toets

Door op de r-toets te drukken wordt het hele spel gereset, zodat het opnieuw gespeeld kan worden.

15. In Game.cpp: de p-toets

Als de p-toets wordt ingedrukt, wordt het spel gepauzeerd. De kip kan niet meer bewegen, en de voertuigen ook niet. Er verschijnt een tekst op een transparante zwarte achtergrond "Pauze". Druk nogmaals op p om de pauze op te heffen en de tekst te verwijderen.

16. Kip animatie

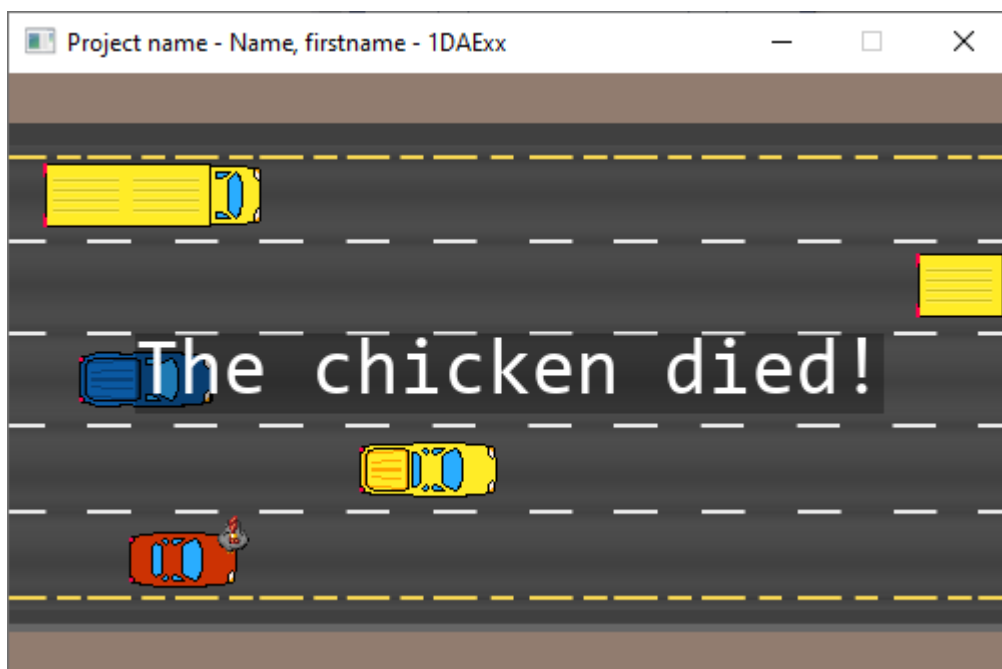
Hou de examentijd in de gaten. Voorkom dat de je reeds opgebouwde functionaliteit niet breekt door deze animatie toe te voegen.

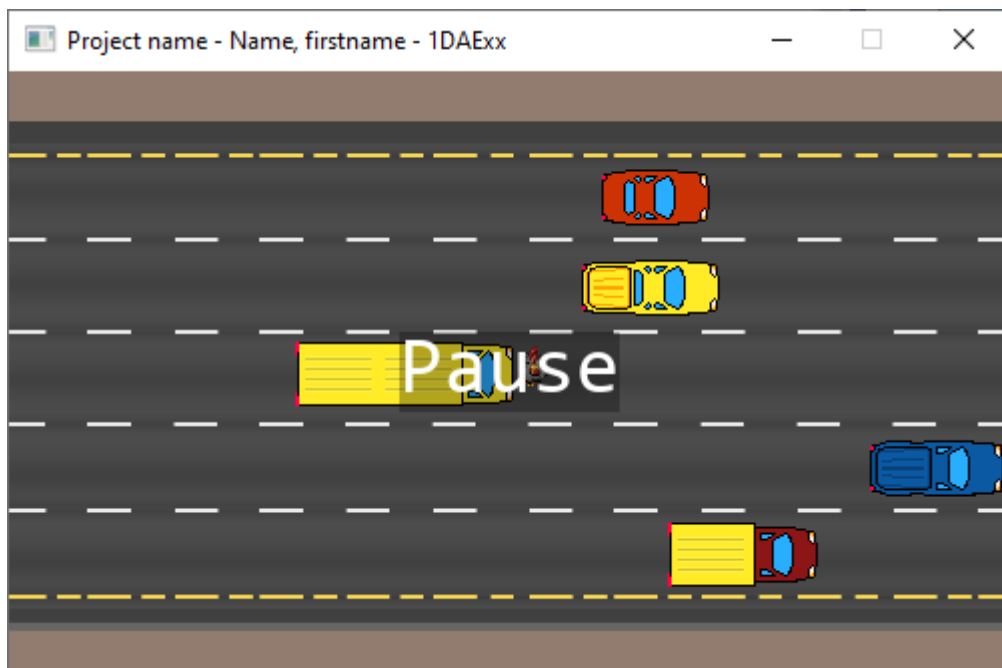
Elk frame van de kip komt overeen met een andere toestand: voor, links en rechts. Gebruik een enumeration om die toestand weer te geven. Gebruik de enumeration state om te beslissen welk frame er getekend moet worden.

17. Voertuig-types

Hou de examentijd in de gaten. Voorkom dat de je reeds opgebouwde functionaliteit niet breekt door deze wijziging toe te voegen.

Maak een willekeurige volgorde van voertuigtypes op de straat. Zorg ervoor dat elk voertuigtype maar één keer getoond wordt. Maak een willekeurige volgorde voor het nummer dat wordt gebruikt bij het maken van voertuigen. Elke keer dat het game wordt gestart, wordt een andere volgorde gegenereerd. Reset wijzigt de voertuig types niet





Screenshot met willekeurige voertuigvolgorde, Spel is in pauze, een fractie van een seconde voordat de kip zou worden geraakt.

Submit

Maak de oplossing schoon. Sluit Visual Studio. Verwijder de .vs en de debug en/of de x64-map.
Controleer nogmaals of de map met de Visual Studio-oplossing alle benodigde code bestanden bevat.
Converteer de oplossingsmap naar een rar/zip-bestand met dezelfde naam als de map.

Dien **dit** rar/zip bestand in onder de leho opdracht van dit lab examen.
Kijk **zelf** of dit correct is gebeurd. Onjuiste examen uploads (bijvoorbeeld een ander project of een project met ontbrekende codebestanden) resulteert in 0 punten.