

C++ periode 4

Stella Rübsaam & Lucas Baneke

Les 1 leerdoelen:

| Leerdoel | Score aan het begin van de les | Score aan het einde van de les | Waarom is het (niet) veranderd? |
|---|--------------------------------|--------------------------------|--|
| Een UML- klassendiagram lezen | 7,5 | 7,5 | nvt |
| Een klassendiagram maken volgens de UML-richtlijnen | 8 | 8 | nvt |
| Een UML- sequence diagram lezen | 7 | 7 | nvt |
| Een sequence diagram maken volgens de UML- richtlijnen | 7,5 | 7,5 | nvt |
| Een C++ project maken aan de hand van UML- diagrammen | 5 | 5 | nvt |
| C++ code compileren | 3 | 5 | Iets geleerd over hoe C++ builden werkt |
| Een eigen gemaakte C++ project uitvoeren | 3 | 5 | ^ |
| Je code structureren volgens de OOP- richtlijnen | 3 | 3 | nvt |
| Werken met klassen in C++ | 6 | 6,5 | Kort gehad over gebruik van klassen |
| Klassen aanmaken in je programmeer omgeving zoals Microsoft Visual Studio | 10 | 10 | nvt |
| Communiceren over code met en beroepsgeenoot | 8 | 8,5 | Heb zojuist met een nieuw iemand dit gedaan, dus = meer ervaring :) |

| | | | |
|---|----|-----|--|
| Presenteren over een code gerelateerd ontwerp | 8 | 8 | nvt |
| Het verschil benoemen tussen een header en een source | 6 | 6 | nvt |
| C++ klassen instantiëren | 6 | 6,5 | iets geleerd over de syntax hiervan |
| C++ functies gebruiken | 6 | 6,5 | Meer soorten functies gezien |
| C++ functies aanmaken | 8 | 8 | nvt |
| Gebruik maken van argumenten in C++ functies | 5 | 5 | nvt |
| Werken met verschillende type arrays zoals vectors | 3 | 3 | nvt |
| Werken met de C++ syntax | 6 | 6 | nvt |
| Input vragen aan de gebruiker | 6 | 6 | nvt |
| Tekst op het scherm laten zien | 10 | 10 | nvt |
| Het verschil benoemen tussen runtime en compile time | 1 | 3 | Research gedaan naar wat dit is en besproken |

Samenvatting briefing periode 4:

Aan het begin van de briefing gaat het over de leerdoelen die we vorige periode hadden en de leerdoelen die we periode 4 willen gaan behalen. Daarna kort de planning van toetsen en vervolgens welke delen van het kwalificatiedossier wij op gaan focussen. Dan is er voor elke week een gedetailleerde beschrijving van wat de bedoeling is en wat er daarvoor allemaal nodig is en ook de daarbijhorende lesdoelen en onderdelen van het kwalificatiedossier.

De eerste week gaat vooral over introductie, het bespreken van de leerdoelen en planning. Week 2 staat in het teken van code bibliotheken en week 3 over die bibliotheken bouwen. In week 4 ga je onder andere kijken naar front-end en back-end in een klassendiagram en in week 5 is er een schriftelijke toets. De week daarop worden de toetsen besproken. Week 7 en 8 gaat over plan van aanpak, Git en reflectie.

Aan het einde zijn nog alle bijlages te vinden met daarin in hoe alles wordt beoordeeld, alle werkbladen en de lesvoorbereidingsformulieren.

Wat je moet inleveren aan het einde

- C++ code wat meer focust op front-end dan back-end
- Documentatie (dit verslag, trello)
- UML (klassendiagram, sequence diagram)

Les 2 bibliotheken (long list):

| Schrijf de naam van de bibliotheek en een cijfer van 1 tot 5 bij de onderwerpen | | | | | |
|--|---------------------|--|--|-------------------------------|--------------------------------------|
| <i>Bibliotheek</i> | <i>Documentatie</i> | <i>Gebruikt door hoeveel programma's</i> | <i>Algemene beschikbaarheid informatie</i> | <i>Toepasbaarheid C++</i> | <i>Overige opmerking/ cijfer</i> |
| SFML | 3 | 4 | 3 | 4 | Info lastig te vinden 3.5 |
| Cinder | 5 | 3 | 4 | 5 | Overzichtelijke website 4.5 |
| Qt | 5 | 5 | 5 | 5 | Veel beginner guides 5 |
| Vulkan | 2 | 3 | 3 | 5 | Onoverzichtelijke documentatie 3 |
| Unreal Engine 4 | 5 | 5 | 5 | 5 | 5 |
| OpenGL | 4 | 3 | 5 | 5 | 4.25 |
| piksel | 2 | 1 | 2 | 5 | 2.5 |
| Stanford Library | 4 | 1 | 4 | 5 | 3.5 |
| CryEngine | 5 | 5 | 5 | 5 | 5 |

Les 3 & 4:

Ik was er 2 lessen niet bij en heb geen tijd besteed aan C++.

Les 5 (short list):

Ik heb de shortlist van Lucas afgemaakt en de streams van de vorige lessen bekeken.

| Bibliotheken | Waarom in de short list? | Categorie 1: Documentatie | Categorie 2: Wiskunde | Categorie 3: Beginner vriendelijk | Categorie 4: Tutorials |
|---------------------|---------------------------------|---------------------------------------|------------------------------------|--|---|
| <i>OpenGL</i> | Grote library met veel opties | Moeilijk te vinden | Geen info beschikbaar | Help | Minimaal |
| <i>SFML</i> | Lijkt fijn voor beginners | Veel info en uitleg van docent | Weinig | Veel tutorials op site. Ook over windows, etc. | Veel |
| <i>Cinder</i> | Nette overzichtelijke website | Goede en overzichtelijke documentatie | Docs voor alle wiskundige functies | "Getting Started" page is er | Third party tutorials zijn moeilijk te vinden |
| <i>Qt</i> | Veel links naar beginner guides | Veel documentatie | QtMath, docs aanwezig | "Getting Started" page is er | Qt site heeft links naar tutorials |
| <i>Vulkan</i> | Lijkt goed met C++ te werken | Heel weinig documentatie | Niet te vinden | Nee, alles is moeilijk te vinden en onbegrijpelijk | Niks gevonden |

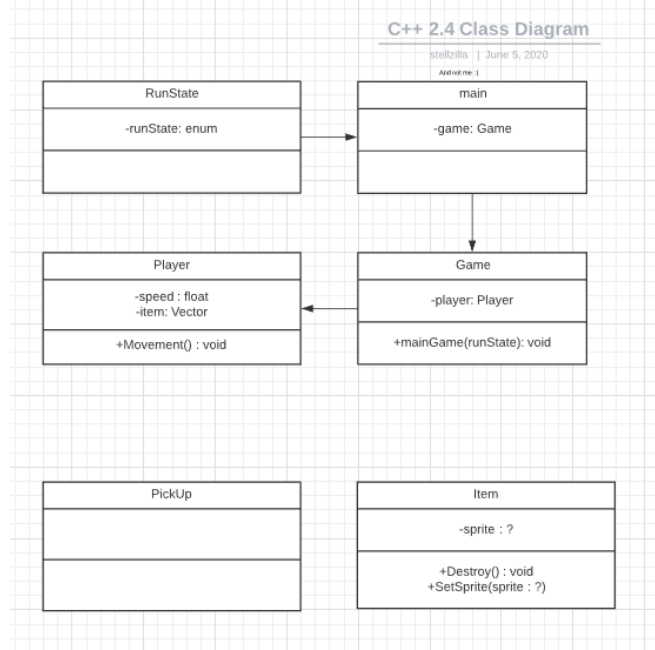
De websites van de libraries zijn heel verwarrend en lijken heel ingewikkeld, maar ik heb er vertrouwen in dat als ik me er in verdiep dat ik dan goede resultaten krijg.

Ik wil graag zo goed als mogelijk C++ leren en dit is daar natuurlijk deel van. Het is ook een leuker onderdeel aangezien je de visuele kant ervan te zien krijgt. Wel vind ik het lastig om volle inzet te geven aan C++ aangezien ik al ongelofelijk veel stress aan mijn hoofd heb door het Mythe project.

Les 6:

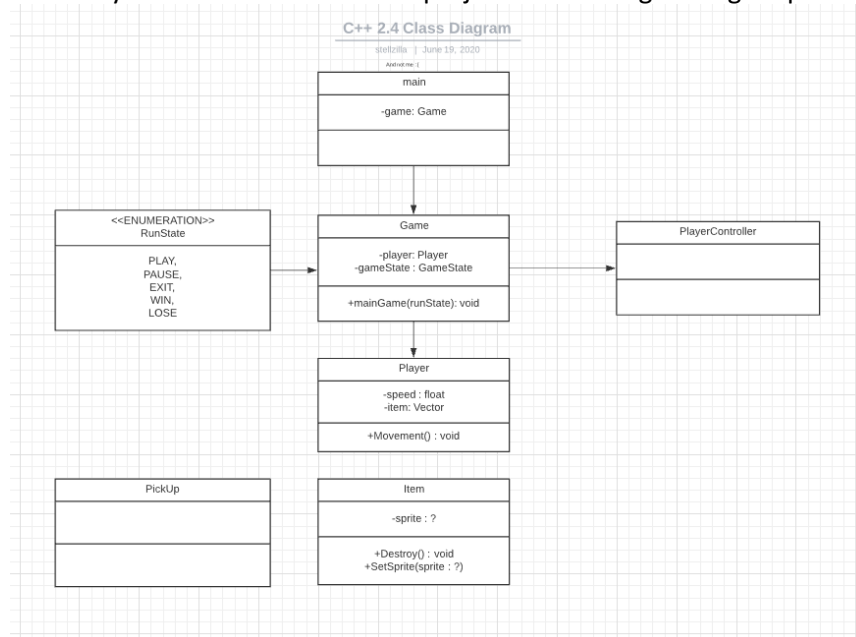
We willen een game maken waarin je rondloopt en dingen op moet pakken. Top down 2D game met inventory system(?). Beginnen met alleen player controls en wanneer je over items loopt, pak je ze op. Versimpelde versie van snake.

We zijn van plan Cinder te gebruiken aangezien de site erg overzichtelijk is. We gaan die bibliotheek zo snel mogelijk werkende krijgen, maar maken eerst de klassendiagram.



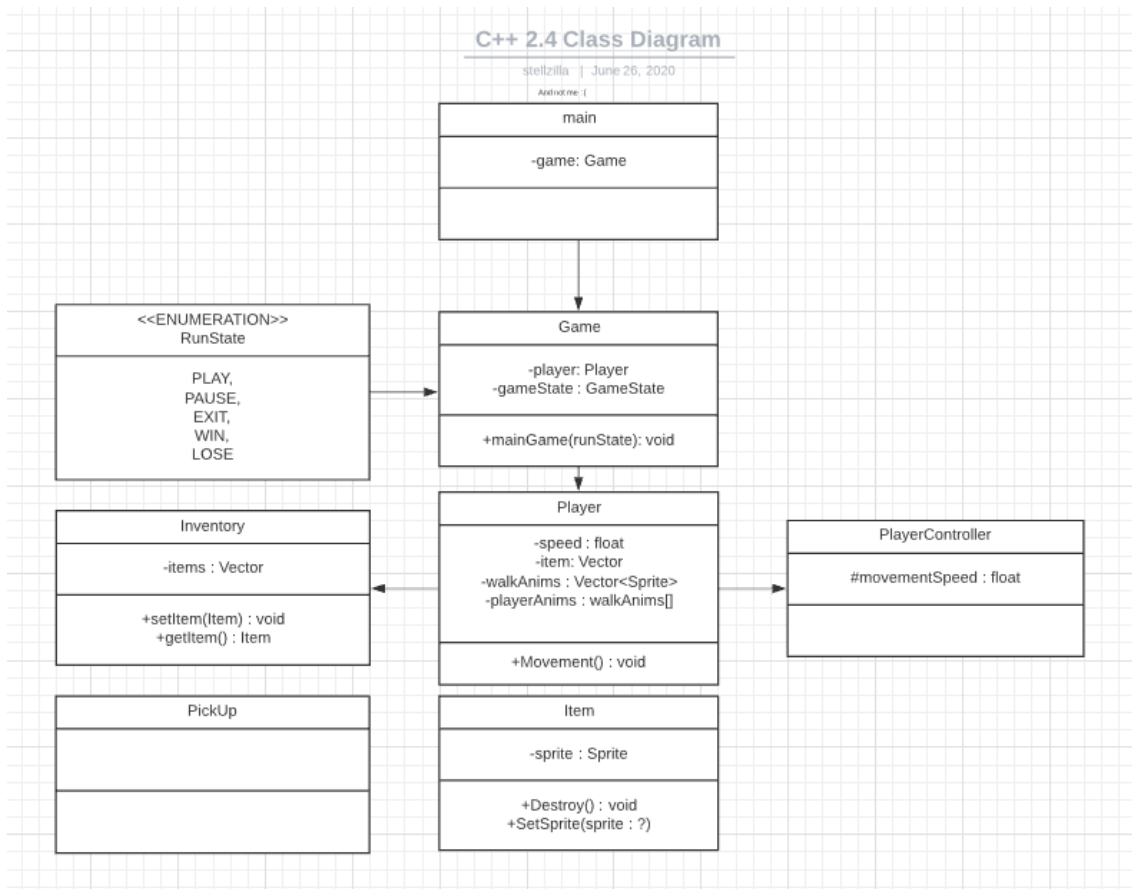
Les ? (idk anymore)

We hebben toch besloten om SFML te gebruiken als library omdat we er niet uitkomen met Cinder. CmakeSettings.txt kunnen we niet vinden in de github source en ook niet in de regular download, cmakeSettings.json is ook nog nergens te bekennen. We proberen de stream te volgen, maar er zijn daar dingen die wij niet hebben in onze source. We hebben alles geregeld in de Visual Studio Installer en vervolgens SFML gedownload. Na het plaatsen van de SFML source code, maakte ik een CmakeSettings json file aan en daarna kon ik de cache maken voor de config. Vervolgens buildde ik de library en linkte ik hem aan het project in de config settings. Update class diagram:



| |
|--|
| Begrip: |
| Benaming van argument |
| Omschrijving |
| Als er een argument bij een functie staat, moet daar ook een naam aan zijn gegeven. |
| Voorbeeld |
| AddObjects(std::vector<Object*> anObject); |
| Begrip: |
| Argument/parameter |
| Omschrijving |
| De variabele die je meegeeft met een functie |
| Voorbeeld |
| void SetPostalCode(std::string aPostalCode); |
| Begrip: |
| Datatype |
| Omschrijving |
| Een soort data die aan de compiler vertelt op welke manier de programmeur de data wil gebruiken. |
| Voorbeeld |
| int, bool, string, etc. |
| Begrip: |
| Return type |
| Omschrijving |
| De waarde die een functie teruggeeft. |
| Voorbeeld |
| public int getShuma() { return Shuma; } |

We hebben een Player in beeld gekregen die kan bewegen door de move functie met de pijltjes knoppen. Daarnaast hebben we een Item in de window gezet en we zijn bezig met een collider achtig systeem te vinden waarmee we de Pickup kunnen laten werken.



Door een array te maken waarin de item wordt gezet, konden wij een loop maken die checkt of de player op de item staat met behulp van de intersects() en getGlobalBounds() functies. Als dat zo is, wordt de item geërased en wordt er gelijk een nieuwe item in de array gepusht met een position tussen de 0 – 800 door de rand() functie.

