

Deel 1: Introductie

Stap 1: Teaser

Object georiënteerd programmeren, wat is:

- Inkapseling (encapsulation)?
- Overerving (inheritance)?
- Veelvormigheid (polymorphism)?

Wat zijn de voordelen hiervan?

Zijn er ook nadelen, zo ja welke?

Wat is het verschil tussen een object en een module?

Stap 2: Wat kun je na deze les

Een programma maken met gebruik van inkapseling, access labels en modules.

Een module hiërarchie opstellen met juist gebruik van headers en *#pragma once*.

Uitleggen wat het verschil is tussen independent en modular compilation.

Uitleggen wat het verschil is tussen een interpreter, een compiler en een linker.

Stap 3: Ophalen benodigde voorkennis

Een object *o'* in C++ is een abstractie van een object *o* in de "echte" wereld, dat met *o* interacteert, *o*'simuleert of gegevens over *o* administreert.

Ook combinaties van deze drie rollen zijn mogelijk.

Praktisch gezien bestaat *o'* uit gegevens (fields) en de functies (methods) die specifiek op die gegevens werken.

Deel 2: Kern

Stap 4: Uitleg van de nieuwe lesstof

Korte samenvatting van de bijbehorende videoles, gelegenheid tot vragen.

Compileren, runnen en bespreken voorbeeldprogramma's.

Uitleg van het begrip analyse-model (domein-model) in tegenstelling tot ontwerp-model.

Stap 5: Voorbeelden en vragen door docent

Geef voorbeelden van objecten uit een realtime besturings-toepassing, plus voorbeelden van relevante fields en methods.

Zelfde vraag voor een simulatie toepassing.

Zelfde vraag voor een administratieve toepassing.

Stap 6: Oefenen met de nieuwe lesstof

Stel een met behulp van ORR-kaarten een domein-model op voor een applicatie voor gebouwenonderhoud.

Een object bestaat uit meerdere bouwdelen, die elk uit secties bestaan.

In elke sectie wordt een bepaald soort programma gedraaid, dat bestaat uit een aantal acties met bijbehorende normen.

Stap 7: Feedback op gemaakte oefening

Een of meerdere leden van elke groep delen hun domein-model, docent en medestudenten geven feedback, evt. na vragen om nadere uitleg.

Deel 3: Afronding

Stap 8: Evalueren of deze les goed “geland” is

Eén of meer leden van elke groep stellen vragen en/of geven tips en/of tops.

Stap 9: Huiswerk om je de lesstof verder eigen te maken

Zie opdrachten-tab in MS-Teams.

Inleveren van de uitgewerkte opdrachten die bij een les horen is, samen met een positief verlopen eind-assessment, een noodzakelijke voorwaarde voor een voldoende en dient uiterlijk 2 volle dagen voor de volgende les plaats te vinden, uitsluitend op de geëigende wijze in MS-Teams. Daarnaast dienen ze in die volgende les voor bespreking beschikbaar te zijn.

De resultaten worden deels in de volgende les, deels individueel besproken. Maak de opdrachten zo goed mogelijk, maar wees ook niet bang om fouten te maken. Het gaat erom dat je een serieuze poging waagt en de docenten je indien nodig kunnen helpen om verder te komen.

--//--