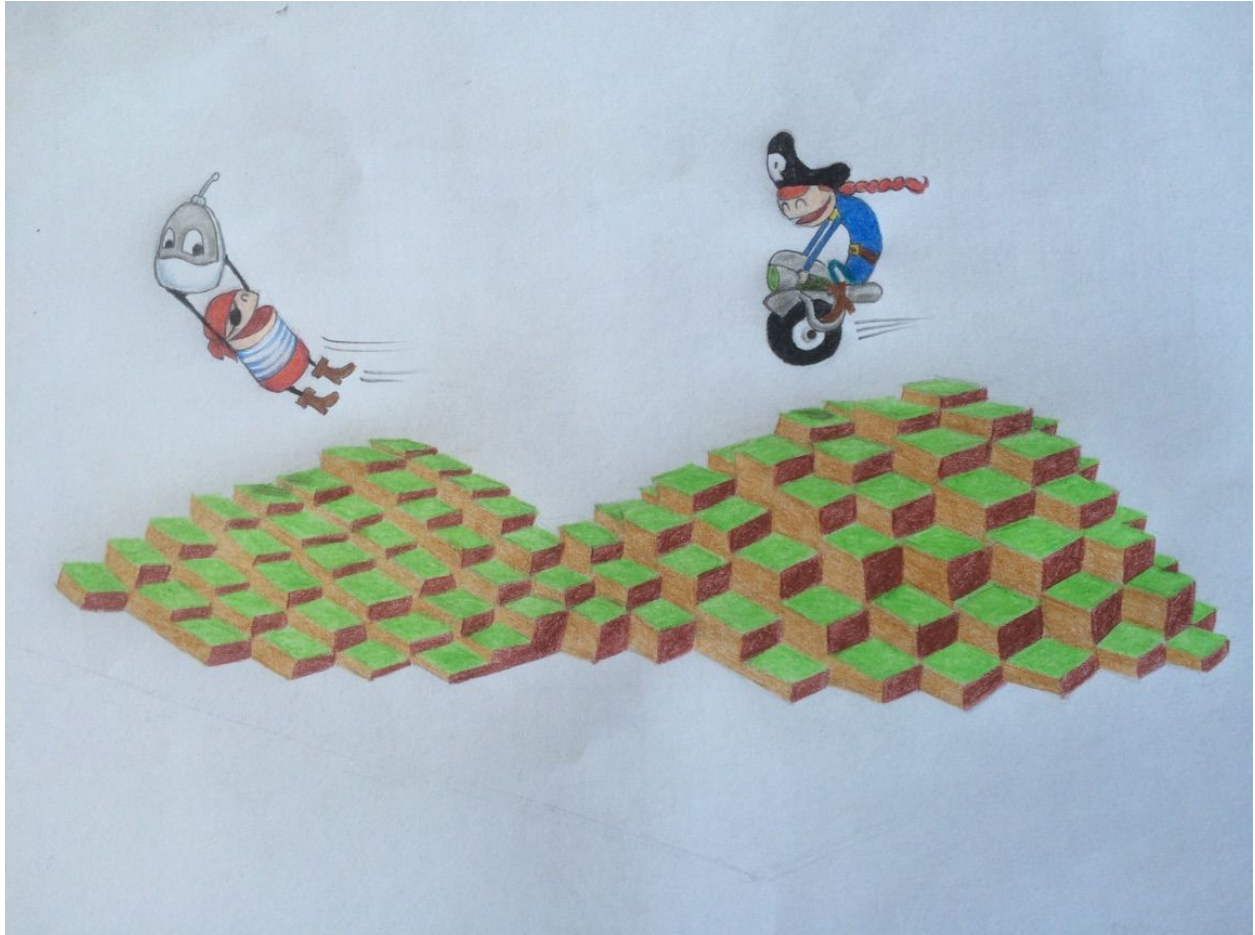


KODU Game Lab



Overskrift Lav 3D-spil med KODU Game Lab

Introduktion Eleverne vil lære de basale funktionaliteter i en introøvelse i brugen af Terræn-editoren og lave et lille program med en KODU der spiser æbler. Eleverne får prøvet kræfter med kunstig intelligens når de skal programmere fodboldspillerne i RoboSoccer. Dernæst vil eleverne lære at lave et af de spil de selv spiller meget i deres fritid, nemlig Tower Defence. Sidst skal eleverne udfordre hinanden i at programmere deres taktikker i det turbaserede multiplayer strategispil "Kampen om Øresund".

Læringsmål

Eleverne lærer at:

- Tænke i processer og algoritmer

- Lave digitale produktioner
- Arbejde med forskellige computerspilsgenrer
- Udvikle abstrakt tænkning
- Anvende visuel programmering
- Navigere i 3D
- Kreativ og innovativ tænkning
- Arbejde problemløsningsorienteret
- Anvende maskinprogrammering og adfærdsprogrammering
- Realisere digitalt produkt

Målgruppe

Alder: 7-10

Elevforudsætninger

Ingen erfaring nødvendig. Dette er et godt sted at starte med programmering og spildesign.

Varighed

i alt 12 lektioner

Intro til terræneditor (1 lektion)

Intro til programmering (1 lektion)

RoboSoccer (2 lektioner)

Tower Defence (6 lektioner)

Slaget om Øresund (2 lektioner)

Undervisningsmetoder

Introøvelserne gennemgås i fællesskab men eleverne har hver deres computer.

De resterende øvelser er i grupper.

Opsætning

KODU Game Lab kan downloades direkte fra kodugamelab.com Det anbefales at underviseren medbringer installationen på en USB-stik (også med det medfølgende XNA-framework) så man slipper for at bruge tid på download (spillet fylder 200 mb) i tilfælde af wifi-problemer.

Hvis dette sikres, vil internet ikke være et behov.

FORLØBET

Intro til terræn-editor (1 lektion)

Først vil eleverne blive sat til at undersøge terræn-editoren. Eleverne skal lære at navigere i kortet, dreje rundt og zoome. Dernæst skal de gøre kortets areal større, lave bjerge/bakker og søer. De skal lege med byggelementerne og lave en sti.

Intro til programmering (1 lektion)

Eleverne skal sætte æbler på kortet ved at trykke på KODU-ikonet og højreklikke på kortet -> add object -> apple. Dernæst skal de tilføje en KODU via samme procedure. Nu skal de programmere KODU'en til at kunne bevæge sig. De introduceres til princippet bag WHEN: ? DO: ? og så skal de prøve selv at programmere KODU'en til at bevæge sig ved piletasterne. Men underviseren skal prøve ikke at sige hvordan det skal gøres og lade dem finde frem til det selv. Løsning: WHEN: keyboard - arrows DO: move

Næste skridt er at få KODU'en til spise æblet, når den støder på det. Igen skal underviseren forsøge ikke at give svaret. Løsning: WHEN: bump- objects - apple DO: eat

Nu får eleverne frit spil til at prøve flere funktioner på egen hånd, såsom color, say, express, jump, m.fl. og udvidde det lille spils objekter og mekanik.

Robo-soccer (2 lektioner)



Formålet er at give eleverne en fornemmelse af hvordan man programmerer adfærd/AI og samtidigt tænke ud af boksen. Eleverne bliver introduceret til kommandoerne bump, see, path, close by, not, quickly, slowly, launch og directions samt funktionen creatables der gør det muligt at oprette kopier af objekter som beholder den kode man har programmeret til dem.

Vi har udviklet en forhåndstemplate på en fodboldbane med mål, måltæller, modstanderens fodboldspillere og en fodbold der respawner i midten af banen når der er scoret. Denne kan downloades her: (

<https://drive.google.com/file/d/0BwSmKAWpoiXejBEb0JDNWUwcVk/edit?usp=sharing>)

Eleverne får nu mulighed for at kigge inde i programmeringen af modstanderens fodboldspillere som ser således ud WHEN: bump - ball - in front DO: turn - toward, WHEN: bump - ball DO: launch - it - strong, WHEN: see - ball - closeby DO: move - toward

- quickly, WHEN: see - ball - closeby - not DO: move - on path - slowly og
WHEN: end of path DO: turn - north

Elevernes opgave er nu at bygge deres eget hold som skal kunne vinde over modstanderne. Det er vigtigt at eleverne får lov til at bruge deres fantasi og kreativitet og underviseren må derfor gerne opfordre dem til at bryde regelsættet i fodbold-spillet og tænke ud af boksen. F.eks. ved at sætte deres fodboldspillere til at skyde mod modstanderen, bygge en stor mur foran eget mål, spawn bolde, osv.

Tower Defence (6 lektioner)

Der skal bygges et TD-spil. Ikke alle eleverne ved hvad man snakker om når man siger "Tower Defence", men så snart man nævner "Bloon's TD" er de med.

Som indledende inspiration ses derfor en youtube-film af spillet, eller der spilles 5 minutters "Bloon's TD" via bloons-towerdefense6.com

Underviseren laver en brainstorm på tavlen med eleverne over de spil-elementer TD indeholder. Trin for trin gennemgås elementerne og der snakkes om hvordan de skrives i KODU.

Elementerne

Creeps/monstre

Disse spawnes fra fjendens base i intervaller af x sekunder. Et creep programmeres ved: WHEN: DO: move - on path og WHEN: bump into - homebase DO: explode - once. Creeps skal også gøres "Creatable" inde i "Change settings" så de kan genkaldes og kopieres af andre elementer.

Tårne

Disse skal ligesom creeps gøres "creatables". Et tårn kan programmeres således WHEN: see - creep - close by DO: shoot - it

En bygger

Spilleren skal kunne bevæge byggeren manuelt ved tastatur eller mus f.eks. WHEN: `press - wasd` DO: `move` . Byggeren kan sættes til at bygge tårnene således WHEN: `click - space` DO: `actions - create - creatables - castle`.

En sti

Stien går fra fjendens base til egen base. Den må gerne være kringlet og labyrint-agtig. Stien laves ved at trykke på ikonet der ligner tre perler på en snor.

Fjendens base

Her spawnes creeps WHEN: `timer - 10 seconds` DO: `create - creatable`

Egen base

Denne har kun den funktion at når den bliver ramt af x antal eksploderende creeps har man tabt. Derfor skriver man WHEN: `hitpoints < 0` DO: `end game`

Bølger

Disse skrives ind i programmeringen på fjendes base, ved at benytte "pages" WHEN: `timer - 60 seconds` DO: `switch page`. En ny side, eller "Page" svarer til en ny adfærd hos det element man programmerer på. Så på næste side sætter man næste bølge igang. Her kan man enten sætte tidsintervallet mellem creep-spawn ned, så de kommer hurtigere WHEN: `timer - 5 seconds` DO: `create - creatable` eller lave nye og stærkere creeps. En kombination af begge kunne også være løsningen. Der er i alt 12 "Pages" og dermed mulighed for 12 bølger.

Penge

Det skal jo ikke være muligt at bygge tårne i en uendelighed. Derfor kan man sætte et pointsystem op, der f.eks. giver spilleren ét point pr. dræbt creep. Det programmeres på creepet WHEN: `hitpoints < 0` DO: `game - score - 1 point - once` og fjerner tre points per bygget tårn. Det programmeres på byggeren WHEN: `keyboard - misc - space` DO: `game - score - subtract - 3 points - once`. På nuværende tidspunkt vil man dog stadig kunne bygge uendeligt med tårne, for det betyder bare at du går i minus på points. Det kan klares ved

at programmere følgende på byggeren WHEN: scored - below - 3 points DO: switch to - page 2 og på anden side programmerer man WHEN: scored - above - 3 points DO: switch to - page 1

Kampen om Øresund (2 lektioner)



Som videreudvikling af både RoboSoccer og Tower Defence kan man afholde “Kampen om Øresund” Konceptet er udviklet af Martin Exner fra Coding Pirates og testet og afprøvet med Magnus Göransson fra CoderDojo i Karlshamn. Her dystes to hold (Danmark og Sverige) om at vinde kontrollen over Øresund. Se video af afholdelsen af Kampen om Øresund her:

<https://www.youtube.com/watch?t=38&v=SdE38CS87WI>

Spillet

Det handler om at vinde over nabolandet ved at programmere sine hære til at angribe fjendens skibe/fly/robotter og indtage fjendens byer.

Opsætning

Scenariet er et kort vi har prædesignet, så det ligner Danmark og Sverige med fire byer på hvert land. Det kan downloades her:

<https://drive.google.com/file/d/0B6T6M1IV7toSbDZ2STIXZWpEY28/view>

I hver by er en havn. Havnen sender skibe ud i retning mod modstanderens byer. Skibene sejer på forprogrammerede ruter og møder midt på ruten fjendens skibe som de så angriber.

Spillet er afbalanceret, således at hvis man intet foretager sig vil ingen vinde.

Når man nærmer sig en fjendehavn og får beskydt den tilstrækkeligt, vil havnen blive erobret og skifte side, således at den nu sender skibe ud tilhørende det andet hold.

Fase #1 (1. iteration)

Holdene skal igang med at forbedre koden på deres hære, så de kan vinde Kampen om Øresund. Det gøres ved at programmere skibe, havne, diverse aktører så de er bedre stillet end modstanderens.

For ikke at det skal blive for let er der følgende benspænd:

- Man må ikke ændre i modstanderens koder
- Man må ikke placere objekter på nabolandet men skal holde sig inden for landets grænser.
- Man må ikke bruge word-editoren til at ændre i verdenen (sætte bjerge op der fungerer som mure, lave landgange/broer, osv.)
- Hvis man overbelaster spillet (og barometret i højre hjørne bliver rødt) så har man brugt for mange objekter og bliver diskvalificeret.

Fase #2 (2. iteration)

Nu sender holdene deres kode til modstanderne og alle modtager derved en version af spillet som nogle modstandere har ændret i. Det er nu holdenes opgave at genvinde slaget ved at gøre egne skibe, havne, osv. endnu bedre end det de forbedringer modstanderen har lavet på egne tropper.

Fase #3 (3. iteration)

Når dette er færdigt vælges ét hold's scenarie ud som sendes til modstanderen en sidste gang. Det er nu alle fra samme hold der skal hjælpe med at vinde en sidste gang og dette slag er det afgørende i Kampen om Øresund.

Det er en god idé at holde de to lande adskilt, så de ikke kan lure koden af fra hinanden men også for at skabe en afstand til fjenden og en holdånd i egne rækker, for at styrke spillets dynamik og nerve.

God fornøjelse! :)