



## Undervisningsbeskrivelse

Termin	Juni 121
Institution	Himmerlands Erhvervs- og Gymnasieuddannelser
Uddannelse	
Fag og niveau	Programmering samlæst -
Lærer	Michael Bohl Jenner (mje)
Hold	2k20

### Forløbsoversigt (5)

Forløb 1	Introduktion til programmering
Forløb 2	Lommeregner
Forløb 3	Tegnsæt og stream
Forløb 4	Grafisk udgave af lommeregner
Forløb 5	Eksamensprojekt

## Forløb 1: Introduktion til programmering

<b>Forløb 1</b>	Introduktion til programmering
<b>Indhold</b>	<p>Introduktion til programmering generelt og til C# sproget. introduktion: kursus fra nemprogrammering.dk, visual studio, namespace, class, data typer, variable, navngivning, type casting, udtryk, operatorer, programstrukturer, kontrolstrukturer (if-else, while-do), metoder, array, strings, rutediagram (første bekendtskab), sandhedstabeller, scope (class, method, codeblock), debugger,</p> <p>Noter:          Installer Visual Studio Community (nyeste).          Følgende lektie er delt op på Windows-brugere (Win) og på Mac brugere (Mac). Bemærk: Nedenstående er individuelle opgaver. Du skal have Visual Studio til at køre på din egen pc, og skal selv navigere rundt i systemet (det er med andre ord ikke tilstrækkeligt at en du plejer arbejde sammen med kan det). Win+Mac: Opret .NET core konsol applikation som skriver "Hello, World!" til skærmen. Åben Solution Explorer og gå lidt på opdagelse her. Hvad ser du? Win: Opret .NET konsol applikation som skriver "Hello, World!" Åben Solution Explorer og gå lidt på opdagelse her. Hvad ser du? Sammenlign med .NET core applikationen. Mac: Find ud hvad eksekverbare filer har som extension på mac? Hvad hedder biblioteksfiler på mac?          Sørg for at du har lavet øvelserne frem til og med "Aritmetiske udtryk i C#", nemprogrammering.dk. (du må gerne arbejde videre frem). Læs som kontrolstrukturer og funktioner i Programmering, Systime - frem til og med "To typer forgreninger". Bogen indeholder desværre ikke C# eksempler, så spring over kode-eksempler og øvelser (du må gerne lave øvelserne i C#).          Lektie = afleveringer.          Gå til yourspace kurset for Programmering, find C# dokumentationen af Microsoft og slå op hvordan man skriver en switch statement. Gem linket til siden i pdf og aflever i uddata - se opgaver.</p>
<b>Omfang</b>	12 lektioner / 12 timer
<b>Særlige fokuspunkter</b>	<p>Fagmål:          læse enkle programmer og redegøre for deres funktionsmåde og anvendelsesmuligheder          rette og tilpasse enkle programmer          anvende eksisterende programdele og biblioteksmoduler i arbejdet med at programmere et fungerende system          løse en enkel problemstilling gennem udviklingen af et program bl.a. i samspil med andre fag          anvende grundlæggende konstruktioner i et programmeringssprog          demonstrere viden om fagets identitet og metoder</p> <p>Kernestof:          programmeringssprog og elementer i programmeringssprogets opbygning såsom data- og kontrolstrukturer          programdele og biblioteksmoduler          abstrakte programmeringsbeskrivelser og dokumentation</p>
<b>Væsentligste arbejdsformer</b>	Individuelle opgaver, pararbejde.

## Forløb 2: Lommeregner

<b>Forløb 2</b>	Lommeregner
<b>Indhold</b>	<p>Her arbejdes med første projekt: Lommeregner konsol app, Opgaver løses i rækkefølge: Opgave 1: Input og output Opgave 2: Metoder Opgave 3: Dynamiske operationer Opgave 4: String parsing Opgave 5: String parsing med string metoder Opgave 6: Potensopløftning med matematikfunktioner Opgave 7: Pi</p> <p>Noter: Gå til: <a href="https://open.kattis.com/">https://open.kattis.com/</a> og opret brugerprofil. Den kommer vi til at bruge i programmeringskurset når vi er færdige med den simple del af lommeregneren - og måske også lidt parallelt med det. Læs hjælpesiden : <a href="https://open.kattis.com/help/csharp">https://open.kattis.com/help/csharp</a> så du ved hvordan du kan submitte en løsning. Besvar derefter denne: <a href="https://open.kattis.com/problems/hello">https://open.kattis.com/problems/hello</a> Se video 21 og løs opgaver til video 21 i nemprogrammering.dk kurset. Opgaverne handler om en vigtig datastruktur i C# som hedder arrays. Dine løsninger afleveres dagen før HER (i pdf format) og der trækkes lod om hvilke løsninger der vises og gennemgås på klassen.</p>
<b>Omfang</b>	4 lektioner / 4 timer
<b>Særlige fokuspunkter</b>	<p>Fagmål: anvende eksisterende programdele og biblioteksmoduler i arbejdet med at programmere et fungerende system løse en enkel problemstilling gennem udviklingen af et program bl.a. i samspil med andre fag anvende grundlæggende konstruktioner i et programmeringssprog demonstrere viden om fagets identitet og metoder</p> <p>Kernestof: programmeringssprog og elementer i programmeringssprogets opbygning såsom data- og kontrolstrukturer programdele og biblioteksmoduler</p>
<b>Væsentligste arbejdsformer</b>	Gruppearbejde.

### Forløb 3: Tegnsæt og stream

<b>Forløb 3</b>	Tegnsæt og stream
<b>Indhold</b>	<p>Tegnsæt, unicode, stream: input-, output- og error-stream, Træning af udarbejdelse af algoritmer til mindre udfordringer på kattis.com platformen.</p> <p>Noter: Se video om at komme igang med Kattis.com - den forklarer blandt andet input-strøm, output-strøm og fejl-strøm (stream) og template som kan anvendes. Løs opgave "Stuck in a time loop" i Kattis.com systemet. Det kan være en fordel at anvende følgende skabelon. NB: Det er en konsol-applikation, .net core. Efter timen samler vi op - og ser hvor mange kattis.com udfordringer i har løst. Det er et af flere input til karaktergivning. Løs mindst en ekstra <a href="http://open.kattis.com">http://open.kattis.com</a> opgave. Jeg har oprettet en liste så i får forskellige open.kattis.com opgaver at arbejde med. Elev-rækkefølgen er random-genereret, men rykket ned så opgaver med står sværhedsgrad undgås. Hver elev skal løse en ny kattis.com opgave. Ideen er at du kigger i ovenstående liste og læser frem til dit navn forekommer første gang. Har du ikke løst opgaven har du fundet din lektie. Har du løst opgaven, fortsætter du nedefter indtil dit navn optræder igen. Har du ikke løst opgaven har du fundet din lektie. Skulle du, mod forventning, have løst begge opgaver så vælger du frit en ny opgave. Brug en rimelig tid på at løse opgaven, f.eks. maks 20 minutter. Dernæst noter hvad der driller og hav det klar til timen. God arbejdslyst Mvh Michael Læs om Array i tavlenote (tilføjelser siden sidst). Læs om List i tavlenote. Løs List-venlig opgave Modulo på open.kattis.com - ved hjælp af List. Udvid Alphabet Spam så den også kan håndtere små bogstaver æøå (lowercase) og store bogstaver ÆØÅ (uppercase). Her er testinput.</p>
<b>Omfang</b>	18 lektioner / 18 timer
<b>Særlige fokuspunkter</b>	<p>Fagmål: læse enkle programmer og redegøre for deres funktionsmåde og anvendelsesmuligheder rette og tilpasse enkle programmer anvende eksisterende programdele og biblioteksmoduler i arbejdet med at programmere et fungerende system demonstrere systematik i programmeringsprocessen løse en enkel problemstilling gennem udviklingen af et program bl.a. i samspil med andre fag anvende grundlæggende konstruktioner i et programmeringssprog demonstrere viden om fagets identitet og metoder</p> <p>Kernestof: programmeringssprog og elementer i programmeringssprogets opbygning såsom data- og kontrolstrukturer programdele og biblioteksmoduler</p>
<b>Væsentligste arbejdsformer</b>	Pararbejde, individuelt arbejde.

## Forløb 4: Grafisk udgave af lommeregner

<b>Forløb 4</b>	Grafisk udgave af lommeregner
<b>Indhold</b>	<p>Grafisk udgave af lommeregner Introduktion til grundlæggende uwp, herunder xaml og databinding.</p> <p>Noter: Vær sikker på at din pc (pånær mac brugere) kan udvikle UWP apps. Det vil sige at UWP pakken er downloadet og at din pc er sat i developer-mode. Vælg et af de seneste lommeregnerprogrammer og fremstil et rute-diagram der beskriver programmet. Når rute-diagrammet er godkendt skal det være en del af din onenote, notesbog om programmering. Software? Her kan f.eks. online diagrams.net anvendes - eller Microsoft Visio hvis du har det.</p> <p>Lav pseudokode og rutediagram (flowchart) til den kode du tidligere har valgt (en af lommeregneropgaverne 4-7). Der er inspiration til pseudokode og rutediagrammer i noterne her og her. Husk at skrive i jeres logbog hvordan det går med at lære pseudokode og rutediagrammer.</p>
<b>Omfang</b>	8 lektioner / 8 timer
<b>Særlige fokuspunkter</b>	<p>Fagmål: læse enkle programmer og redegøre for deres funktionsmåde og anvendelsesmuligheder rette og tilpasse enkle programmer anvende eksisterende programdele og biblioteksmoduler i arbejdet med at programmere et fungerende system løse en enkel problemstilling gennem udviklingen af et program bl.a. i samspil med andre fag anvende grundlæggende konstruktioner i et programmeringssprog demonstrere viden om fagets identitet og metoder</p> <p>Kernestof: programmeringssprog og elementer i programmeringssprogets opbygning såsom data- og kontrolstrukturer programdele og biblioteksmoduler</p>
<b>Væsentligste arbejdsformer</b>	Projektarbejde.

## Forløb 5: Eksamensprojekt

<b>Forløb 5</b>	Eksamensprojekt
<b>Indhold</b>	<p>Projekt opgave som ligner et rigtigt eksamensforløb. Projekt oplæg præsenteres, projekt beskrivelser udarbejdes, efter godkendelse kan arbejdet påbegyndes. Perioden inkluderer afsluttende præsentation og forsvar af projekter.</p> <p>Noter: Svar på spørgeskema om ønsker / ideer til eksamensprojekt. For "afsluttede" er det det rigtige eksamensprojekt, for "fortsættere" er det et øve-projekt. Gå til Teams, Programmering og vælg klassenotesbogen, her skal du skrive i siden "LogBog" - hvis du ikke er startet endnu så udfyld mindst en af de 4 første: 1. Introduktion, 2. Lommeregner, 3. open.kattis.com, 4. Grafisk brugerinterface. Hvis du er startet udfyldes mindst en mere.</p>
<b>Omfang</b>	24 lektioner / 24 timer
<b>Særlige fokuspunkter</b>	<p>Fagmål: læse enkle programmer og redegøre for deres funktionsmåde og anvendelsesmuligheder rette og tilpasse enkle programmer anvende eksisterende programdele og biblioteksmoduler i arbejdet med at programmere et fungerende system demonstrere systematik i programmeringsprocessen løse en enkel problemstilling gennem udviklingen af et program bl.a. i samspil med andre fag anvende grundlæggende konstruktioner i et programmeringssprog demonstrere viden om fagets identitet og metoder</p> <p>Kernestof: programmeringssprog og elementer i programmeringssprogets opbygning såsom data- og kontrolstrukturer programdele og biblioteksmoduler arbejdsgange og systematik i programmeringsprocessen abstrakte programmeringsbeskrivelser og dokumentation</p>
<b>Væsentligste arbejdsformer</b>	Gruppearbejde.



## Undervisningsbeskrivelse

<b>Termin</b>	June 2022
<b>Institution</b>	Himmerlands Erhvervs- og Gymnasieuddannelser
<b>Uddannelse</b>	vaf
<b>Fag og niveau</b>	Programmering B
<b>Lærer</b>	Michael Bohl Jenner (mje)
<b>Hold</b>	3k21

### Forløbsoversigt (5)

<b>Forløb 1</b>	Spilplatformen Unity
<b>Forløb 2</b>	Projekt: Tic Tac Toe
<b>Forløb 3</b>	Projekt: Tower of Hanoi
<b>Forløb 4</b>	Eksamensprojekt
<b>Forløb 5</b>	Afslutning

## Forløb 1: Spilplatformen Unity

Forløb 1	Spilplatformen Unity
----------	----------------------



<p><b>Indhold (1/2)</b></p>	<p>Udarbejdelse af simple scripts i Unity, herunder repetition af metodebegrebet.</p> <p>Udarbejdelse af simpelt adventure-spil.</p> <p>Derefter lærerstyret indførelse i block-breaker spil-udvikling, sprites, gameobjekter (klasse og objekt-begrebet), vektorer i struct, Klassediagram som dokumentationsform.</p> <p>Noter:</p> <p>Vi skal bruge værktøjerne Visual Studio Community fra Microsoft og Unity. Installer følgende i rækkefølge: Visual Studio (har du den allerede, så start programmet og opdater den ... opdateres ofte). Unity hub (så man kan håndtere forskellige Unity udgaver). Inde fra Unity Hub, under Installs, tilføjes en Unity udgave (en relativ ny LTS = long term udgave foretrækkes).</p> <p>Mindst en NumberWizard med Debug.Log og Input.GetKeyDown("1"); Mindst en NumberWizardUI - med mindst en scene - og besked i konsol ved klik på knap.</p> <p>Læs om metoder i 3.1.2 til og med 3.1.5 i Yellow Book om C-sharp på engelsk.</p> <p>Lektie: Skriv mindst 30 minutter på dit adventure spil. I får 1 time til adventure spil, derefter gå vi videre med bold og blokke der skydes ned.</p> <p>I BlockBreaker spillet sørger du for at man kan hoppe mellem de 3 scener - ved at anvende SceneLoader game-objekt og et SceneLoader script. Opret folder kaldet Sprites og tilføj filerne fra yourspace. Træk dem ind på scene hvorved de bliver til gameobjects - og udfør det sidste der står i word filen her.</p> <p>Forbered huskelegen - her er de mulige spørgsmål.</p> <p>Test indhold i Huskelegen. Vedr Vector2 i Unity. Man erklærer en 2D vektor med "Vector2 minVektor;". Dernæst kan man finde på at skrive "minVektor = new Vector2(...); Hvad gør de to instrukser ovenfor (erklæring og new ...)?</p> <p>Lektien er at du har et fungerende grundlæggende Block Breaker spil. Det vil sige: to scener til to baner (Level 1 og Level 2), at du har lavet det meste til prefabs (EventSystem skal ikke ændres) - så du hurtigt kan bygge Level 2 - efter Level 1 er færdig og stort set alle elementer er lavet til prefabs (se word dokument for hints). Bold følger paddle, venstre klik på musen gør at bolden slippes. Når bolden falder forbi paddle går du til game over siden. Når bold rammer Block så slettes Block (destroy). I timerne gør vi spillet "færdigt", dvs. vi tæller antal ramte blokke, og hvis alle blokke er ramt går man til næste level.</p> <p>Vi nåede ikke helt i mål i går - så lektien til i dag er at få talt breakableBlocks korrekt op for hver scene, at få scoreCount talt korrekt - og vist på Canvas (hint: indsæt UI, Text - og forbind det til script så script kan holde værdien ved lige - lige som vi gjorde i NumberWizardUI). Derudover er lektien at overveje et spil som du selv bygger i Unity. 2D og 3D er begge i orden.</p> <p>At have tegnet et klassediagram med GameObjekterne: Ball, Paddle, Lose-Collider, Block, Level, GameSession, SceneLoader. At forstå hvert eneste tegn i alle .cs filerne (i det udleverede projekt). Måske bliver du hørt i dem. Hav særligt fokus på point-systemet. Når et spil skifter scene så nedlægges en masse objekter og nye oprettes. Hvordan "bæres" point med over?</p> <p>Projektbeskrivelse afleveres efter første time.</p> <p>Det meste af timerne går med projekt. Men samtidig hæver vi lige baren lidt for huskelegen, se "Huskelegen Konstruktioner" - så den komm-</p>
-----------------------------	---

<b>Indhold (2/2)</b>	<p>er vi også til at køre. I har nu god tid til at forberede svar, send endelig spørgsmål hvis et punkt i den nye huskeleg er uklart.</p> <p>Forbered huskelegen. Brug 20 minutter på projekt.</p> <p>Forbered huskelegen. Brug 20 minutter på projekt.</p> <p>Forbered demo at jeres system. Varighed cirka 5 minutter per gruppelem (derfor 15 minutter for typisk 3 mandsgruppe). Indhold: Demo system - hvordan er det blevet til? først gjorde vi ... så ... Hvert gruppelem griber fat i stump kode og forklarer linje for linje.</p>
<b>Omfang</b>	35 lektioner / 35 timer
<b>Væsentligste arbejdsformer</b>	Lærerstyret undervisning, med efterfølgende elev-gentagelse. Individuelt arbejde. Pararbejde.

## Forløb 2: Projekt: Tic Tac Toe

<b>Forløb 2</b>	Projekt: Tic Tac Toe
<b>Indhold</b>	Projekt tic tac toe Udvikling af konsol spil. Træning i analyse af rutediagram.
<b>Omfang</b>	4 lektioner / 4 timer
<b>Væsentligste arbejdsformer</b>	Lærerstyret undervisning, projektarbejde.

### Forløb 3: Projekt: Tower of Hanoi

<b>Forløb 3</b>	Projekt: Tower of Hanoi
<b>Indhold</b>	<p>Projektarbejde: Tower of Hanoi Udvikling af konsol applikation.</p> <p>Ret fejl i ukendt kode, forklare kodens flow i form af flow chart, (over- ordnet flow),</p> <p>Introduktion til rekursive metoder, Med fakultetsbegr- ebet. Med follow me sessioner. Fibonacci kæden, to bagud. - sammenli- gnede rekursion med iteration.</p> <p>Fik towers of hanoi engine, tilpass- es. Towers of hanoi, Konsolapplikation. - krav: hver desk er en klas- se med constructors. - bagved 3 stakke med push og pop.</p> <p>Noter: Færdiggør flowchart for Tower of Hanoi Engine</p>
<b>Omfang</b>	20 lektioner / 20 timer
<b>Væsentligste arbejdsformer</b>	Lærerstyret undervisning, individuelt arbejde, pararbejde.

## Forløb 4: Eksamensprojekt

<b>Forløb 4</b>	Eksamensprojekt
<b>Indhold</b>	Eksamensprojekt.
<b>Omfang</b>	24 lektioner / 24 timer
<b>Væsentligste arbejdsformer</b>	

## Forløb 5: Afslutning

<b>Forløb 5</b>	Afslutning
<b>Indhold</b>	<p>Afslutning, repetition og gennemgang af eksamensform.</p> <p>Noter: Kør de 3 første quiz'er færdigt (1 fra w3school og 2 fra tutorialsteacher). Lav note med hvilke fejl du begik - de skal fejres - så er der potentiale til at lære noget - og derefter skal de granskes så begrebet forstås så fejlen ikke begås igen. Jeg lærte f.eks. at struct er en value-type og ikke en reference-type da jeg kørte quiz 1 fra tutorialsteacher.com.</p>
<b>Omfang</b>	3 lektioner / 3 timer
<b>Væsentligste arbejdsformer</b>	