

## Undervisningsbeskrivelse

<b>Termin</b>	Aug 2019 – maj 2021
<b>Institution</b>	HTX Silkeborg, College360
<b>Uddannelse</b>	HTX
<b>Fag og niveau</b>	Programmering niveau B
<b>Lærer(e)</b>	Gorm Drachmann
<b>Hold</b>	htx318prob

<b>Titel 1</b>	Introduktion til programmering / kode
<b>Indhold</b>	Mindre program i App Lab der overordnet giver eleverne en indsigt i hvad kodning er.
<b>Omfang</b>	6 lektioner
<b>Særlige fokuspunkter</b>	<b>Faglige mål:</b> - bruge programmering til at undersøge et emne eller problemområde, med henblik på – via programmets funktion - at skabe ny indsigt eller til at løse et problem. <b>Kernestof:</b> - programmeringssprog og elementer i programmers opbygning, herunder variabler, typer, udtryk, kontrolstrukturer, parametrisering / abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre. - arkitekturen for programmers interaktion med omgivelserne med henblik på hændelsesstyret interaktion og interaktion mellem systemer
<b>Væsentligste arbejdsformer</b>	Undervisning med video og opgaveløsning.

<b>Titel 2</b>	Emne: Grundlæggende forståelse af koden i C#
<b>Indhold</b>	<p>Grundlæggende C# via konsol</p> <p>Tager udgangspunkt i Nemprogrammering.dk</p> <p><a href="https://www.nemprogrammering.dk/Tutorials/c-sharp/programmerings-kursus.php">https://www.nemprogrammering.dk/Tutorials/c-sharp/programmerings-kursus.php</a></p>
<b>Omfang</b>	<p>Ugerne 34 – 52 / 2018</p> <p>36 lektioner</p> <p>Elevtid: ca. 20 timer</p> <p><b>Områder</b></p> <ol style="list-style-type: none"> <li>1 Introduktion til C#</li> <li>2 Kommentarer og udskrift til konsol</li> <li>3 Variabler</li> <li>4 Datatyper (int, float, double og char)</li> <li>5 Strings (tekst-streng)</li> <li>6 Aritmetiske udtryk</li> <li>7 Variabler i udtryk</li> <li>8 Syntaktisk sukker og klistre plus</li> <li>9 Boolean variabler</li> <li>10 Kontrolstruktur ( if else )</li> <li>11 Switch statements</li> <li>12 Loops (while, for og do-while)</li> <li>13 Udvidet kontrolstruktur</li> <li>14 Objektorienteret programmering i C#</li> <li>15 Metoder og instanser</li> <li>16 Returtyper og parametre</li> <li>17 Instansvariabler</li> <li>18 Referencer</li> </ol>

	<p>19 Nedarvning</p> <p>20 Konstruktors</p> <p>21 Arrays</p> <p>22 Gennemløb af arrays</p> <p>23 Public, Private og Protected</p> <p>24 Lav et program med C#   En lommeregner, del 1/3</p> <p>25 Lav et program med C#   En lommeregner, del 2/3</p> <p>26 Lav et program med C#   En lommeregner, del 3/3</p>
<b>Særlige fokuspunkter</b>	<p><b>Faglige mål:</b></p> <ul style="list-style-type: none"> <li>- bruge programmering til at undersøge et emne eller problemområde, med henblik på – via programmets funktion - at skabe ny indsigt eller til at løse et problem.</li> </ul> <p><b>Kernestof:</b></p> <ul style="list-style-type: none"> <li>- programmeringssprog og elementer i programmets opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering / abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre.</li> <li>- arkitekturen for programmets interaktion med omgivelserne med henblik på hændelsesstyret interaktion og interaktion mellem systemer</li> </ul>
<b>Væsentligste arbejdsformer</b>	Videobaseret undervisning med tilhørende individuel opgaveløsning.

<b>Titel 3</b>	Reverse Engineering – analyse af program/kode
<b>Indhold</b>	Eleverne fik udleveret et program som de så skulle analysere og forklare opbygningen, strukturen og koden.
<b>Omfang</b>	4 moduler af 2x45 minutter
<b>Særlige fokuspunkter</b>	<p><b>Faglige mål</b></p> <p>Eleverne skal kunne:</p> <ul style="list-style-type: none"> <li>–bruge programmering til at undersøge et emne eller problemområde, med henblik på – via programmets funktion - at skabe ny indsigt eller til at løse et problem</li> <li>–anvende avancerede konstruktioner i et programmeringssprog</li> </ul>

	<p>–redegøre for arkitekturen af programmer på forskellige abstraktionsniveauer, herunder relationen mellem brug og funktion</p> <p>–rette, tilpasse og udvide avancerede programmer</p> <p>–demonstrere viden om fagets identitet og metoder</p> <p>–arbejde inkrementelt og systematisk i programmeringsprocessen.</p> <p><b>Kernestof</b></p> <p>Gennem kernestoffet skal eleverne opnå faglig fordybelse, viden og kundskaber.</p> <p>Kernestoffet er:</p> <p>–programmeringssprog og elementer i programmers opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre</p> <p>–arkitekturen for programmers interaktion med omgivelserne med henblik på hændelsesstyret interaktion og interaktion mellem systemer</p> <p>–generiske programdele og biblioteksmoduler</p>
<b>Væsentligste arbejdsformer</b>	Klasseundervisning/anvendelse af fagprogrammer/skriftligt arbejde/eksperimentelt arbejde

<b>Titel 4</b>	Database
<b>Indhold</b>	<p>Brug Microsoft SQL Server Management Studio i forbindelse med brugen af database. Der vil være fokus på følgende:</p> <ul style="list-style-type: none"> <li>- Oprette tabeller</li> <li>- Hente og gemme data</li> <li>- Hente og gemme data vha VS</li> <li>- Connection strings</li> <li>- Filtrering af data ud fra SQL og lambda funktioner</li> <li>- ER diagrammer</li> <li>- Relationel database</li> </ul> <p>Perioden afsluttes med en opgave hvor eleverne skal lave et program der går ind og bruger en lokal database eller database på server. Dokumentation der beskriver database.</p>

<b>Omfang</b>	30 moduler, svarende til 45 klokketimer
<b>Særlige fokuspunkter</b>	<p>- bruge programmering til at undersøge et emne eller problemområde, med henblik på – via programmets funktion - at skabe ny indsigt eller til at løse et problem</p> <p>–anvende avancerede konstruktioner i et programmeringssprog</p> <p>–redegøre for arkitekturen af programmer på forskellige abstraktionsniveauer, herunder relationen mellem brug og funktion</p> <p>–redegøre for simple specifikationsmodeller og realisere disse i simple velstrukturerede programmer samt teste disse</p> <p>–rette, tilpasse og udvide avancerede programmer</p> <p>–demonstrere viden om fagets identitet og metoder</p> <p>–arbejde inkrementelt og systematisk i programmeringsprocessen.</p>
<b>Væsentligste arbejdsformer</b>	<ul style="list-style-type: none"> <li>- Klasseundervisning</li> <li>- Øvelsesarbejde, individuelt og i små grupper</li> <li>- Præsentationer i grupper</li> </ul> <p>Eleverne vil skulle lave en evaluering af de andres opgaver og præsentation af kode til de andre elever.</p>

<b>Titel 5</b>	WEB programmering – client side
<b>Indhold</b>	<p>Fokusområder:</p> <ul style="list-style-type: none"> <li>- Grundlæggende HTML</li> <li>- Tabeller, formularer, og medier</li> <li>- CSS – ovenbygning med bootstrap eller andet</li> <li>- Javascript</li> <li>- Webservices, performance og sikkerhed</li> <li>- App med webprogrammering - “Phonegap Build”</li> </ul> <p>Faget tager udgangspunkt i det stof der er gennemgået i Kom IT og bygger ovenpå.</p>
<b>Omfang</b>	30 moduler, svarende til 45 klokketimer

<b>Særlige fokuspunkter</b>	-bruge programmering til at undersøge et emne eller problemområde, med henblik på – via programmets funktion - at skabe ny indsigt eller til at løse et problem
<b>Væsentligste arbejdsformer</b>	<ul style="list-style-type: none"> <li>- Klasseundervisning</li> <li>- Øvelsesarbejde, individuelt og i små grupper</li> <li>- Præsentationer i grupper</li> </ul> <p>Eleverne vil skulle lave en evaluering af de andres opgaver og præsentation af kode til de andre elever.</p>

<b>Titel 6</b>	WEB programmering – server side
<b>Indhold</b>	<p>Fokusområder:</p> <ul style="list-style-type: none"> <li>- .net core</li> <li>- Databaser</li> <li>- Sikkerhed</li> </ul> <p>Perioden afsluttes med en hjemmeside der forbinder client side med serverside med en API kommunikation der gør brug af database delen.</p>
<b>Omfang</b>	30 moduler, svarende til 45 klokketimer
<b>Særlige fokuspunkter</b>	<ul style="list-style-type: none"> <li>- bruge programmering til at undersøge et emne eller problemområde, med henblik på – via programmets funktion - at skabe ny indsigt eller til at løse et problem</li> <li>–behandle problemstillinger i samspil med andre fag</li> <li>–anvende avancerede konstruktioner i et programmeringssprog</li> <li>–redegøre for arkitekturen af programmer på forskellige abstraktionsniveauer, herunder relationen mellem brug og funktion</li> </ul>

	<p>–redegøre for simple specifikationsmodeller og realisere disse i simple velstrukturerede programmer samt teste disse</p> <p>–rette, tilpasse og udvide avancerede programmer</p> <p>–demonstrere viden om fagets identitet og metoder</p> <p>–arbejde inkrementelt og systematisk i programmeringsprocessen.</p>
<b>Væsentligste arbejdsformer</b>	<ul style="list-style-type: none"> <li>- Klasseundervisning</li> <li>- Øvelsesarbejde, individuelt og i små grupper</li> <li>- Præsentationer i grupper</li> </ul> <p>Eleverne vil skulle lave en evaluering af de andres opgaver og præsentation af kode til de andre elever.</p>

<b>Titel 7</b>	Før eksamens opgave
<b>Indhold</b>	Et produkt der vil svare til en eksamenopgave
<b>Omfang</b>	13 moduler, svarende til 19,5 klokketimer.
<b>Særlige fokuspunkter</b>	Se afsnit for eksamensprojekt
<b>Væsentligste arbejdsformer</b>	Se afsnit for eksamensprojekt

<b>Titel 8</b>	Eksamensprojekt
<b>Indhold</b>	<p>I den afsluttende periode af undervisningen afsættes 20 timers undervisningstid til, at eleverne med vejledning fra læreren, udarbejder et eksamensprojekt i grupper på to til tre. Hvor dette ikke er muligt eller ønskeligt, kan man lade eleverne arbejde individuelt.</p> <p>Eksamensprojektet består af et produkt samt en synopsis. Synopsen skal dokumentere udviklingen af det færdige produkt og må højst have et omfang af fem til otte normalsider, eksklusiv koder, rutediagrammer, bilag mm. Eksamensprojektet</p>

	<p>udarbejdes inden for rammerne af projektoplæg stillet af skolen.  Eksamensprojektgrupperne udarbejder en fælles projektbeskrivelse, der inkluderer en beskrivelse af den enkelte eksaminands fokus. Projektbeskrivelsen godkendes af skolen, når beskrivelsen er tilstrækkelig fagligt bred og niveaumæssigt relevant.  Eksamensprojektets synopsis er individuelt udarbejdet.</p>
<b>Omfang</b>	13 moduler, svarende til 19,5 klokketimer.
<b>Særlige fokuspunkter</b>	<p>-Eleverne skal kunne:</p> <ul style="list-style-type: none"> <li>-bruge programmering til at undersøge et emne eller problemområde, med henblik på – via programmets funktion - at skabe ny indsigt eller til at løse et problem</li> <li>-behandle problemstillinger i samspil med andre fag</li> <li>-anvende avancerede konstruktioner i et programmeringssprog</li> <li>-redegøre for arkitekturen af programmer på forskellige abstraktionsniveauer, herunder relationen mellem brug og funktion</li> <li>-redegøre for simple specifikationsmodeller og realisere disse i simple velstrukturerede programmer samt teste disse</li> <li>-rette, tilpasse og udvide avancerede programmer</li> <li>-demonstrere viden om fagets identitet og metoder</li> <li>-arbejde inkrementelt og systematisk i programmeringsprocessen.</li> </ul>
<b>Væsentligste arbejdsformer</b>	<p>Produkter:</p> <ul style="list-style-type: none"> <li>- Projekt rapport + program.</li> </ul> <p>Arbejdsformer:</p> <ul style="list-style-type: none"> <li>- Projektarbejde, individuelt eller i små grupper.</li> </ul>



