



Undervisningsbeskrivelse

Termin	June 2024
Institution	Himmerlands Erhvervs- og Gymnasieuddannelser
Uddannelse	
Fag og niveau	Programmering samlæst -
Lærer	Michael Bohl Jenner (mje)
Hold	3kg23 Pro

Forløbsoversigt (12)

Forløb 1	Websserver og SQL
Forløb 2	Algoritmemønstre og pseudokode
Forløb 3	Projekt - Frit valg
Forløb 4	Miniforløb om rekursion
Forløb 5	Miniforløb om værdityper og referencetyper
Forløb 6	Projekt - Frit valg
Forløb 7	Miniforløb om polymorfi
Forløb 8	Projekt med hardware
Forløb 9	Opsamlingsperiode
Forløb 10	Projekt - frit valg
Forløb 11	Eksamensprojekt
Forløb 12	Afslutning

Forløb 1: Webserver og SQL

Forløb 1	Webserver og SQL
Indhold	<p>Tværfagligt forløb med Kom/IT: Stilladseret udvikling af webserver med SQL til web-shop. SQL database server SQLite. Introduktion til SQL kommandoer. SQL dataadgang fra C#. HTTP server med C# / .NET. U- dlevering af .html fil med embedded javascript som foretager http queries.</p> <p>Noter: Se video om databaser og kort introduktion til SQL: https://www.nemprogrammering.dk/Tutorials/MySQL/MySQLi/1-introduktion.php (vi kommer ikke til at bruge PHP) Prøv et par SQL kørsler i w3schools online tutorial om SQL. Vort mål er at nå frem til SQL Count, Avg,, Sum sektionen og ikke længere. Træn lidt SQL på w3school (vi arbejder frem imod Count, Avg, Sum). Prøv at udføre SQL kommando i DB Explorer på din egen database. Den simpleste de fleste starter med er "select * from <tabelnavn>" indsæt dit tabelnavn istedet for <tabelnavn>. Sidst oprettede vi filer til læsning til og skrivning fra SQLite database (fil). Hvis du ikke kom helt i mål kan du se et løsningsforslag her. Gennemgå Main metoden i Program.cs (se ovenstående zip fil). Kan du fortolke hvad koden gør? Bed evt chatbot om hjælp til forklaring af hvad koden gør. Læs introduktion til Javascript på w3schools. Vi skal kun kradse i overfladen ifht Javascript så gå ikke op i detaljerne. Forklar index.html filen uploadet til yourspace. Anvend chat-bot til at forstå html og javascript. Målet med nedenstående øvelser er at undersøge hvordan WebAPItest01 virker. Installer .NET 7 frameworket. .NET 6 virker ikke med Web API. Installer WebAPItest01 ved følgende: Find folderen hvor din solution er installeret (kan evt ses i Visual Studio inden man går ind i solution). Hvis du har kaldt din solution det samme som jeg har hedder den DatabaseAPI. Åben fileexplorer (finder på mac), og download WebAPItest01, kopier den til folderen og pak den ud (unzip). Åben din solution (DatabaseAPI) i Visual Studio. Højreklik på solution, og tilføj eksisterende projekt, og peg på .csproj filen i WebAPItest01 folderen. Undersøg virkemåden af koden - du må bruge alle de hjælpemidler du kan finde på.</p>
Omfang	9 lektioner / 9 timer

<p>Særlige fokuspunkter</p>	<p>Fagmål: bruge programmering til at undersøge et emne eller problemområde, med henblik på ζ via programmets funktion - at skabe ny indsigt eller til at løse et problem behandle problemstillinger i samspil med andre fag anvende avancerede konstruktioner i et programmeringssprog demonstrere viden om fagets identitet og metoder</p> <p>Kernestof: programmeringssprog og elementer i programmers opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre arkitekturen for programmers interaktion med omgivelserne med henblik på hændelsesstyret interaktion og interaktion mellem systemer generiske programdele og biblioteksmoduler</p>
<p>Væsentligste arbejdsformer</p>	<p>Lærerstyret undervisning. Individuel implementation.</p>

Førløb 2: Algoritmestrukturer og pseudokode

Førløb 2	Algoritmestrukturer og pseudokode
Indhold	Introduktion af simple computationelle problemer såsom: sorter et array af heltal. Løsningsprincip: insertion-sort. Pseudokode som implementerer insertion sort. Implementation som foretager insertion sort. F-rit valg af sorteringsalgoritme og efterfølgende afprøvning.
Omfang	2 lektioner / 2 timer
Særlige fokuspunkter	Fagmål: bruge programmering til at undersøge et emne eller problemområde, med henblik på ϵ via programmets funktion - at skabe ny indsigt eller til at løse et problem anvende avancerede konstruktioner i et programmeringssprog redegøre for simple specifikationsmodeller og realisere disse i simple velstrukturerede programmer samt teste disse arbejde inkrementelt og systematisk i programmeringsprocessen Kernestof: programmeringssprog og elementer i programmerens opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmestrukturer arbejdsgange og systematik i programmeringsprocessen, herunder test og fejlfinding abstrakte programmeringsbeskrivelser og dokumentation
Væsentligste arbejdsformer	Lærerstyret undervisning. Individuelt med parvis sparring. Use-Modify-Create.

Forløb 3: Projekt - Frit valg

Forløb 3	Projekt - Frit valg
Indhold	Projekt - frit valg. Noter: Undersøg: Computere kan kun lagre tal, hvordan lagres tegn?
Omfang	8 lektioner / 8 timer
Særlige fokuspunkter	<p>Fagmål: bruge programmering til at undersøge et emne eller problemområde, med henblik på λ via programmets funktion - at skabe ny indsigt eller til at løse et problem behandle problemstillinger i samspil med andre fag anvende avancerede konstruktioner i et programmeringssprog redegøre for arkitekturen af programmer på forskellige abstraktionsniveauer, herunder relationen mellem brug og funktion redegøre for simple specifikationsmodeller og realisere disse i simple velstrukturerede programmer samt teste disse rette, tilpasse og udvide avancerede programmer demonstrere viden om fagets identitet og metoder arbejde inkrementelt og systematisk i programmeringsprocessen</p> <p>Kernestof: programmeringssprog og elementer i programmers opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre arkitekturen for programmers interaktion med omgivelserne med henblik på hændelsesstyret interaktion og interaktion mellem systemer generiske programdele og biblioteksmoduler arbejdsgange og systematik i programmeringsprocessen, herunder test og fejlfinding abstrakte programmeringsbeskrivelser og dokumentation</p>
Væsentligste arbejdsformer	Gruppearbejde.

Forløb 4: Miniforløb om rekursion

Forløb 4	Miniforløb om rekursion
Indhold	Rekursion Rekursive metodekald. Call stack Nedtælling. Mappegennemløb. Towers of Hanoi. Noter: Løs en ekstra opgave i forløbet om rekursion. Løs endnu en opgave i forløbet om rekursion.
Omfang	5 lektioner / 5 timer
Særlige fokuspunkter	Fagmål: rette, tilpasse og udvide avancerede programmer demonstrere viden om fagets identitet og metoder Kernestof: programmeringssprog og elementer i programmets opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre generiske programdele og biblioteksmoduler
Væsentligste arbejdsformer	Lærerstyret undervisning. Individuelt med parvis sparring.

Førløb 5: Miniforløb om værdityper og referencetyper

Førløb 5	Miniforløb om værdityper og referencetyper
Indhold	Værdityper Referencetyper Metode-argumenter: værdityper. Metode-argumenter: referencetyper Streng er immutable. Struct versus Class.
Omfang	4 lektioner / 4 timer
Særlige fokuspunkter	Fagmål: anvende avancerede konstruktioner i et programmeringssprog rette, tilpasse og udvide avancerede programmer demonstrere viden om fagets identitet og metoder Kernestof: programmeringssprog og elementer i programmets opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre generiske programdele og biblioteksmoduler
Væsentligste arbejdsformer	Parvis arbejde. Use-modify-create

Forløb 6: Projekt - Frit valg

Forløb 6	Projekt - Frit valg
Indhold	<p>Udarbejdelse af produkt i grupper. Ramme: Frit valg.</p> <p>Noter: Kontrollér at din klassenotesbog er opdateret med note om Sortering i sektionen Logbog. Repeter opbygning af computer. Repeter model for arbejdshukommelse.</p>
Omfang	12 lektioner / 12 timer
Særlige fokuspunkter	<p>Fagmål: bruge programmering til at undersøge et emne eller problemområde, med henblik på λ via programmets funktion - at skabe ny indsigt eller til at løse et problem behandle problemstillinger i samspil med andre fag anvende avancerede konstruktioner i et programmeringssprog redegøre for arkitekturen af programmer på forskellige abstraktionsniveauer, herunder relationen mellem brug og funktion redegøre for simple specifikationsmodeller og realisere disse i simple velstrukturerede programmer samt teste disse rette, tilpasse og udvide avancerede programmer demonstrere viden om fagets identitet og metoder arbejde inkrementelt og systematisk i programmeringsprocessen</p> <p>Kernestof: programmeringssprog og elementer i programmets opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre arkitekturen for programmets interaktion med omgivelserne med henblik på hændelsesstyret interaktion og interaktion mellem systemer generiske programdele og biblioteksmoduler arbejdsgange og systematik i programmeringsprocessen, herunder test og fejlfinding abstrakte programmeringsbeskrivelser og dokumentation</p>
Væsentligste arbejdsformer	<p>Gruppearbejde. Inkrementel udvikling.</p>

Førløb 7: Miniførløb om polymorfi

Førløb 7	Miniførløb om polymorfi
Indhold	Introduktion Compile-time polymorfi (metode overloading) Runtime polymorfi (metode overriding ved nedarvning og polymorfi ved interface).
Omfang	6 lektioner / 6 timer
Særlige fokuspunkter	Fagmål: anvende avancerede konstruktioner i et programmeringssprog rette, tilpasse og udvide avancerede programmer Kernestof: programmeringssprog og elementer i programmets opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre generiske programdele og biblioteksmoduler
Væsentligste arbejdsformer	Parvis selvstudie.

Forløb 8: Projekt med hardware

Forløb 8	Projekt med hardware
Indhold	Projekt med benspænd. Hardware skal indgå (en eller flere af: arduino, esp, dobot robot-arm).
Omfang	13 lektioner / 13 timer
Særlige fokuspunkter	<p>Fagmål:</p> <p>bruge programmering til at undersøge et emne eller problemområde, med henblik på ζ via programmets funktion - at skabe ny indsigt eller til at løse et problem</p> <p>behandle problemstillinger i samspil med andre fag</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> <p>Kernestof:</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> <p>arbejdsgange og systematik i programmeringsprocessen, herunder test og fejlfinding</p> <p>abstrakte programmeringsbeskrivelser og dokumentation</p>
Væsentligste arbejdsformer	Gruppearbejde.

Forløb 9: Opsamlingsperiode

Forløb 9	Opsamlingsperiode
Indhold	Opsamling på diverse inden næst-sidste projekt. Eleverne arbejder selvstændigt på at fjerne mangler.
Omfang	7 lektioner / 7 timer
Særlige fokuspunkter	Fagmål: anvende avancerede konstruktioner i et programmeringssprog redegøre for simple specifikationsmodeller og realisere disse i simple velstrukturerede programmer samt teste disse rette, tilpasse og udvide avancerede programmer demonstrere viden om fagets identitet og metoder Kernestof: programmeringssprog og elementer i programmets opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre generiske programdele og biblioteksmoduler
Væsentligste arbejdsformer	Individuelt arbejde. Parvis sparring.

Forløb 10: Projekt - frit valg

Forløb 10	Projekt - frit valg
Indhold	Projekt - emne: frit valg.
Omfang	14 lektioner / 14 timer
Særlige fokuspunkter	<p>Fagmål: 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 behandle problemstillinger i samspil med andre fag anvende avancerede konstruktioner i et programmeringssprog redegøre for arkitekturen af programmer på forskellige abstraktionsniveauer, herunder relationen mellem brug og funktion redegøre for simple specifikationsmodeller og realisere disse i simple velstrukturerede programmer samt teste disse rette, tilpasse og udvide avancerede programmer demonstrere viden om fagets identitet og metoder arbejde inkrementelt og systematisk i programmeringsprocessen</p> <p>Kernestof: programmeringssprog og elementer i programmets opbygning, herunder variable, typer, udtryk, kontrolstrukturer, parametrisering/abstraktionsmekanismer, rekursion, polymorfi og algoritmemønstre arkitekturen for programmets interaktion med omgivelserne med henblik på hændelsesstyret interaktion og interaktion mellem systemer generiske programdele og biblioteksmoduler arbejdsgange og systematik i programmeringsprocessen, herunder test og fejlfinding abstrakte programmeringsbeskrivelser og dokumentation</p>
Væsentligste arbejdsformer	Gruppearbejde

Forløb 11: Eksamensprojekt

Forløb 11	Eksamensprojekt
Indhold	Eksamensprojekt. Noter: Skim opdateret gymdok f.eks. om versionsstyring og samarbejde mm - i bunden af siden.
Omfang	21 lektioner / 21 timer
Særlige fokuspunkter	<p>Fagmål:</p> <p>bruge programmering til at undersøge et emne eller problemområde, med henblik på λ via programmets funktion - at skabe ny indsigt eller til at løse et problem</p> <p>behandle problemstillinger i samspil med andre fag</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> <p>Kernestof:</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> <p>arbejdsgange og systematik i programmeringsprocessen, herunder test og fejlfinding</p> <p>abstrakte programmeringsbeskrivelser og dokumentation</p>
Væsentligste arbejdsformer	Gruppearbejde (produkt). Individuelt (synopsis)

Forløb 12: Afslutning

Forløb 12	Afslutning
Omfang	2 lektioner / 2 timer
Væsentligste arbejdsformer	



Undervisningsbeskrivelse

Termin	June 2023
Institution	Himmerlands Erhvervs- og Gymnasieuddannelser
Uddannelse	
Fag og niveau	Programmering samlæst -
Lærer	Michael Bohl Jenner (mje)
Hold	2g22 pro

Forløbsoversigt (7)

Forløb 1	Introduktion til programmering
Forløb 2	Datastrømme stdin, stdout, stderr
Forløb 3	GUI programmering med C# og .NET MAUI
Forløb 4	Projekt .NET MAUI applikation
Forløb 5	Datastrukturer
Forløb 6	Snake projekt
Forløb 7	Eksamensprojekt

Førløb 1: Introduktion til programmering

Førløb 1	Introduktion til programmering
Indhold	<p>Ser videoer om programmering i C#. Efter hver video er der tilhørende opgaver. Læs note om computermodel. Læs note om Model for hukommelse, lagring af tal og tekst og forskellene. Læs note om metoder og løse tilhørende opgaver.</p> <p>Noter: Tjek at du kan logge på moodle (aka yourspace). Download Visual Studio Community 2022. Vi undersøger sproget C# ved at følge kompendiet her: https://sites.google.com/view/c-sharp-kompendium/start/1-introduktion-til-c Løs alle øvelser til sektion 2 i C# kompendiet. Hav dem klar så du kan fremvise på forespørgsel. Løs øvelserne i sektion 2 og 3 om variable og data typer. Læs om model af computerens arbejdshukommelse frem til og med side 8.</p> <p>Kan du, med egne ord, forklare begreberne: 1. variabel og data type? Hvordan erklæres variabel? Tildeling af værdi? Giv tre forskellige eksempler. 2. boolske variable? og deres anvendelse? 3. hvad er en string? anvendelse? 4. kontrol-strukturen "if - else"? Giv et eksempel. 5. kontrol-strukturen "while"? Giv et eksempel. Skriv note med dine svar. I har nu arbejdet metoder via kompendiet. Læs udleveret note om metoder og løs opgave 1-4.</p>
Omfang	13 lektioner / 13 timer
Særlige fokuspunkter	<p>Fagmål: læs enkle programmer og redegøre for deres funktionsmåde og anvendelsesmuligheder rette og tilpasse enkle programmer</p> <p>Kernestof: programmeringssprog og elementer i programmeringssprogets opbygning såsom data- og kontrolstrukturer programdele og biblioteksmoduler</p>
Væsentligste arbejdsformer	Individuelt og parvist. Lærer viser programmering af udvalgte løsninger (live programmering),

Forløb 2: Datastrømme stdin, stdout, stderr

Forløb 2	Datastrømme stdin, stdout, stderr
Indhold	<p>Datastrømmene stdin, stdout, stderr introduceres og der løses udvalgte opgaver hos tjenesten http://open.kattis.com/ Dokumentations- og udviklingsformer og metoder. Inline kommentarer. Rutediagrammer. Ps-eudokode.</p> <p>Noter: Se video om at bruge kattis online træningsværktøjet i denne sektion. Opret konto på open.kattis.com og prøv at lave dit første script (se skabelon på yourspace). Repetér rutediagrammer ved at skimme tavlenoter fra sidst. Kan du lave et rutediagram for en for-løkke? På papir eller eventuelt på computer. Sørg for at du som minimum har løst K attis udfordringen "Hello World" så eventuelle opstarts-problemer er løst. Se evt de sidste to sider i slides fra video om at anvende Kattis - der er hint til at løse kompileringsproblemer. Når vi skal løse opgaver i kattis støder vi ofte på behovet for at anvende Array. Lektien er derfor at se video 21 og video 22 om array. Ingen denne gang.</p>
Omfang	8 lektioner / 8 timer
Særlige fokuspunkter	<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>
Væsentligste arbejdsformer	Individuel og parvis. Lærer viser udvalgte eksempler via live programmering.

Forløb 3: GUI programmering med C# og .NET MAUI

Forløb 3	GUI programmering med C# og .NET MAUI
Indhold	<p>GUI programmering med C# og .NET MAUI. XAML sproget til udformning af GUI. Med fokus på GRID. Event i GUI. Eventhåndtering.</p> <p>Noter: Opdater Visual Studio til nyeste, dvs. 17.3.6 (eller nyere). Installer desuden Workload NET MAUI. Hvis du er i tvivl om hvordan du tilføjer workload, så kig med i video 2 herunder. Microsoft har lavet en lille video-række på 8 videoer om at komme i gang med .NET MAUI. De er på engelsk, så her får du trænet lidt engelsk - hvilket også er en del af fagets bekendtgørelse. Det kan måske være en hjælp at slå auto-genererede undertekster til (engelske). Den første video forklarer hvad .NET MAUI er - se den her (engelsk). Den næste video forklarer installation - her behøver vi kun at køre Windows eller Mac, ikke Android eller iOS (men interesserede kan naturligvis prøve det). Videoen er her. Den tredje video giver et overblik over de mange filer der er i et .NET MAUI projekt. Bemærk: der er mange filer og du behøver ikke kunne overskue dem alle, men prøv at køre programmet på din platform (Win / Mac) og prøv at ændre tekst-størrelse som James Montemagno også gør i videoen. Videoen er her. Se Mosh's video om Grid layout. Afprøv koden mens du ser videoen, sæt video på pause, afprøv, se lidt mere, afprøv. Videoen er den bedste jeg har kunnet finde, men passer til Xamarin XAML dialekten. Men med en lille justering - se yourspace - så virker det præcist i NET MAUI's XAML dialekt. Efter du har set videoen kan du svare på: Hvad betyder det når man anvender * som højde for en række? Hvad betyder det når man anvender 80 som bredde for en søjle? Hvad betyder det når man anvender Auto som bredde for en søjle? Hav et groft layout for drikkepenge-beregneren klar.</p>
Omfang	6 lektioner / 6 timer
Særlige fokuspunkter	<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 abstrakte programmeringsbeskrivelser og dokumentation</p>
Væsentligste arbejdsformer	Lærer introducerer .NET MAUI biblioteket og dets muligheder. Eksemplarisk eksempel vises :drikkepenge-beregner.

Forløb 4: Projekt .NET MAUI applikation

Forløb 4	Projekt .NET MAUI applikation
Indhold	Eleverne arbejder i grupper med selvvalgt applikation. Ramme: Det skal være .NET MAUI grafisk app.
Omfang	9 lektioner / 9 timer
Særlige fokuspunkter	<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>
Væsentligste arbejdsformer	Gruppearbejde. Præsentation af udviklet produkt.

Forløb 5: Datastrukturer

Forløb 5	Datastrukturer
Indhold	<p>Til løsning af mere komplicerede opgaver har vi brug for et bibliotek af værktøjer: Data strukturer (og algoritmer til håndtering af dem).</p> <p>Datastrukturer: Array (repetition), Lists, Hash tabel (dictionary), Stak og Kø.</p> <p>Noter: Repetition: Hvad er syntaksen for et simpelt konsol C# program? (instrukser i ...). Repetition: Hvordan skriver jeg en if og en if-else sætning? Repetition: Hvordan skriver jeg en for, while og en do-while løkke? Løs opgave 1 og 2 fra noten om datastrukturer i afsnit "Opgaver om array".</p>
Omfang	7 lektioner / 7 timer
Særlige fokuspunkter	<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 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>
Væsentligste arbejdsformer	Individuelt og parvis.

Forløb 6: Snake projekt

Forløb 6	Snake projekt
Indhold	<p>Snake projekt. Lærer viser ufærdigt program-skelet som anvender game loop, timer og lister til snake-krop og hash tabeller til kollisionsdetektion. Eleverne arbejder derefter videre med koden og foretager forbedringer.</p> <p>Noter: Overvej hvordan man kan lave en timer i game loopet i snake. Timeren skal kunne spawnne mad eller forhindringer på tilfældige tidspunkter, f.eks. hvert 15. sekunder +/- 4 sekunder.</p>
Omfang	8 lektioner / 8 timer
Særlige fokuspunkter	<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>
Væsentligste arbejdsformer	Lærerstyret undervisning. Gruppearbejde.

Forløb 7: Eksamensøveprojekt

Forløb 7	Eksamensprojekt
Indhold	Ud fra projektoplæg udarbejdes projektbeskrivelse til godkendelse. Deretter udarbejdes synopsis og løsning i form af C# solution. Bemærk: For B elever er dette et øve-eksamensprojekt.
Omfang	16 lektioner / 16 timer
Væsentligste arbejdsformer	Projektarbejde.