# C# Datahåndtering

|  |  |
| --- | --- |
| **Læringsmål** | Du kan:   * **1Pf1**: anvende centrale metoder til at specificere og konstruere algoritmer [...] * **1Pf2**: anvende centrale faciliteter i programmeringssproget til realisering af algoritmer […] * **1Pf3**: anvende et i professionen udbredt, integreret udviklingsværktøj, herunder versionsstyringssystem […] til at designe og konstruere praksisnære applikationer […] * **1Pk3**: i en struktureret sammenhæng tilegne sig ny viden, færdigheder og kompetencer inden for programmeringssprog, udviklingsværktøjer, programmeringsteknikker og programdesign |
| **Forventede  produkter** | * Implementering af programmeringsøvelserne |
| **Forventet læringsudbytte [SOLO]** | Designsporet:   * [Unistrukturel] Du kan identificere enkelte kendetegn ved **Computational Thinking** – dekomponering, mønstergenkendelse (en: pattern recognition)   Programmeringssporet:   * [Unistrukturel] Du kan genkende enkelte kendetegn ved C# **datatyper** – int, string, bool, double, char * [Unistrukturel] Du kan nævne enkelte kendetegn i **datahåndtering** – variable, virkefelter, udtryk og operatorer i disse samt tildeling (en: assignment) og tilhørende tildelings­operatorer   Færdighedssporet:   * [Unistrukturel] Du kan genkende enkelte kendetegn ved **versionsstyring** (en: version control) – GitLab login, Git Bash, Git-kommandoer: init, status, add, commit og push |
| **Din forberedelse** | Designsporet:   * [Computational Thinking: Pattern Recognition](https://www.youtube.com/watch?v=SixLnIDV1yY) (video: kun de første 7:29) * [Pattern Recognition (BBC)](https://www.bbc.co.uk/bitesize/guides/zxxbgk7/revision/1) (3 sider + Test og Score)   Programmeringssporet:   * [An introduction to object and classes](https://www.youtube.com/watch?v=tWIe9E4SWQo) (video: 8:45) * [YB] 2.1 - 2.2 (s. 14-33) * [Operators, Expression, Statements](https://channel9.msdn.com/Series/CSharp-Fundamentals-for-Absolute-Beginners/Operators-Expressions-and-Statements) (video: 15:22) * [Floating points](https://csharp.net-tutorials.com/data-types/floating-points/) * [C# Character Data Type](http://www.blackwasp.co.uk/CSharpCharacterDataType.aspx) (se bort fra nullable character; det kommer vi til) * [String](https://www.youtube.com/watch?v=Gw9kLUhV0dQ) (video: 2:18) * [Working With Strings](https://code-maze.com/csharp-basics-string-methods/)   Færdighedssporet:   * [Introduction to Git](https://www.notion.so/Introduction-to-Git-ac396a0697704709a12b6a0e545db049) (til og med punkt 4.4) * [Learn Git in 15 minutes](https://www.youtube.com/watch?v=USjZcfj8yxE) (video: 00:00 – 10:45)   + Fokus på git kommandoer: config, init, status, add, commit, push, log   + staging og repository |

Du skal i denne opgave se videre på datahåndtering med udtryk og operatorer, introduceres til to nye datatyper double og char samt klasser. Derudover skal du se på mønstergenkendelse i CT og begynde med Git. **Sørg for at gøre øvelse 3 og 4 i Ex02-Conditionals færdig som forberedelse, da de vil indgå i denne opgave.**

# Dagens ord: Mønstre og objekter

I dagligdagen oplever vi mange slags mønstre, både bevidst og ubevidst, og oftest ser vi sammenhænge mellem forskellige objekter, vi ser hver dag, f.eks. at forskellige objekter også kan have fælles egenskaber. Selvom to bygninger sjældent er helt ens, så vil der altid være visse fællestræk, som kendetegner en bygning (tag, døre, vinduer, etc.). Det samme gælder for en bil, en cykel, osv. Selve opgaven at finde de træk, der er fælles, og dem som er forskellige, er netop tanken bag mønstergenkendelse i Computational Thinking. Dette skal du arbejde med i denne opgave samt med et par ekstra datatyper, beregninger og GIT.