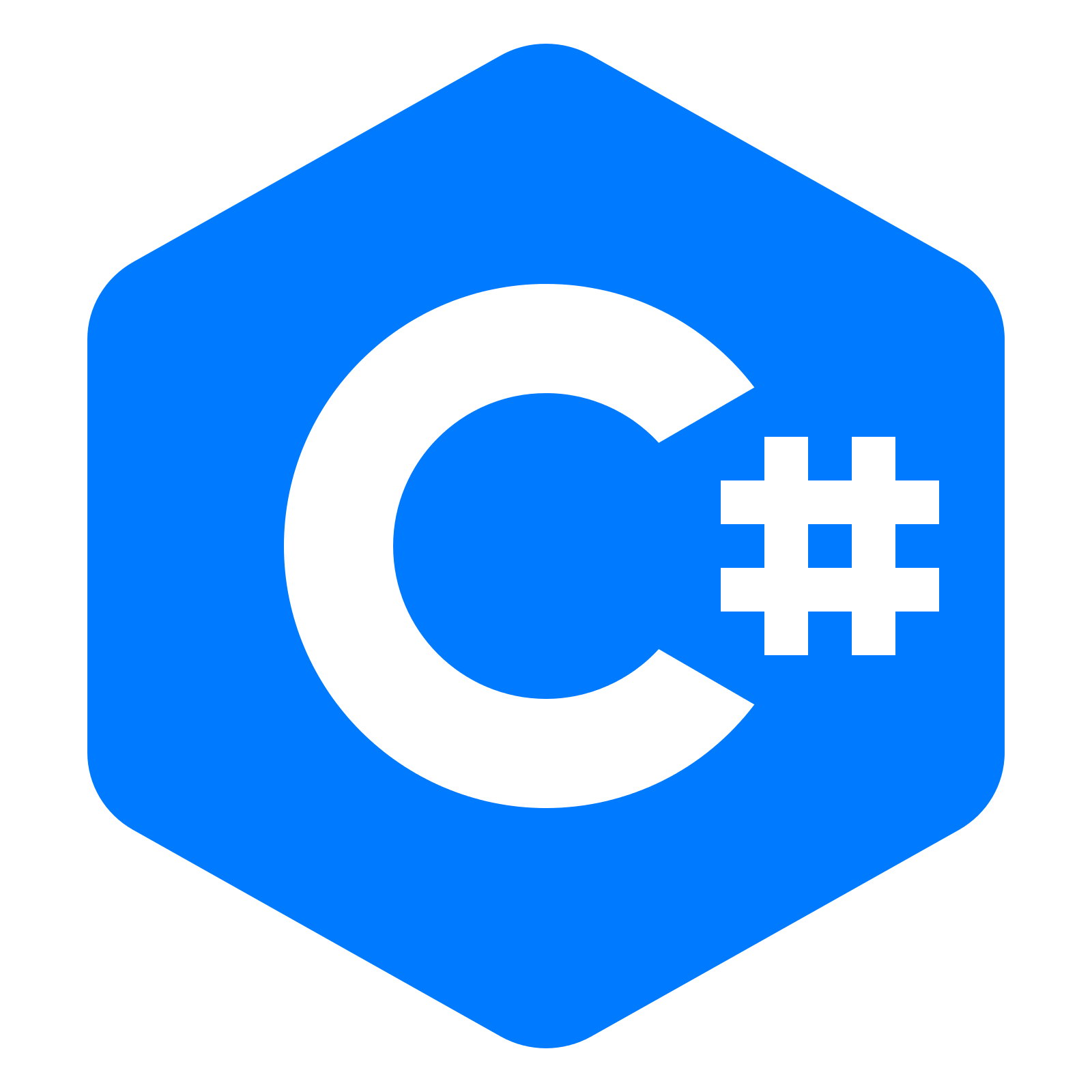
Obligatoriske Programmeringsopgave Bytebuster

Af Magnus Refsgaard Christoffersen, TEC D54



# Indholdfortegnelse

Side 1) Formål og kavspecifikationer

Side 2) Ind- og uddata plus problemer og fejl

Side 3) Brugermanual

Side 4) Konklusion og vedhæftede billag

# Formålet med bytebuster

Formålet med den obligatoriske opgave var at lave et kommandoprompt program i C# efter anvisning fra elevstudiebogen på side 11-13. Christian Mørk leder efter nye talenter og har brug for et program vi skal lave.

# Kravspecifikationer

* personen kan højst få 3 rigtige eller 3 forkert.
* Programmet skal have en velkommenhilsen, taber besked og vinder besked.
* ByteBuster kan udskrive 5 tilfældige binært tal.
* programmet kan stoppe ved max 3 rigtige og max 3 forkerte, eller 5 hvis den rammer 5 forskellige udskrivelser, som er maksimum.

# Ind- og uddata

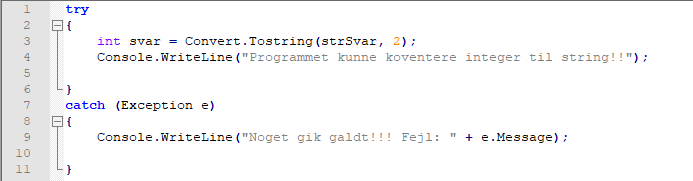
Disse ord er simple termininologi, al data brugeren skriver ind i konsollen, textfeltet og mus forandring er inddata/input. Mens uddata/output er al data computeren udskriver.

Her har vi et eksemple på ind- og uddata. Uddata er blå, mens inddata er rød.



# Problemer og fejl

En af de mest normale fejl i programmet er at brugeren måske skriver bogstaver ind i stedet for tal, så konverteringen fejler siden a, b, z, \*, ¤ ikke er tal. Vi kan bruge try & catch til at tjekke om inputtet er tal. dette try & catch kan prøve at køre koden uden at gå helt istå hvis der går noget galdt.

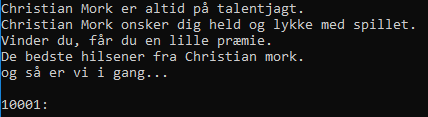


Programmet kan ikke skelne mellem et forkert svar og ingen svar. Vi kan nemt implemetere kort kode der kan tjekke om der overhoved står noget i vores svar variable.

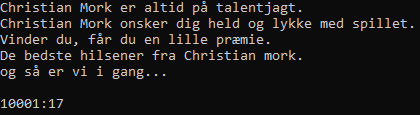
# Brugermanual

Brugermanualen beskriver skridt for skridt, på hvordan brugeren skal bruge programmet og hvordan programmet fungere (test systemet hvor programmet kører på er desværre ikke på dansk, og har derfor problemer med at udskrive bogstaver i det dansk-norske alfabet.)

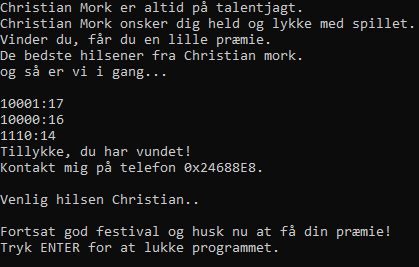
1. Her får brugeren en velkomst besked og hvad han/hun skal gøre.



1. Nu kan brugeren indstaste et svar ind i programmet. Dette gør brugeren indtil hun eller han for enden 3 forkerte eller 3 rigtige.

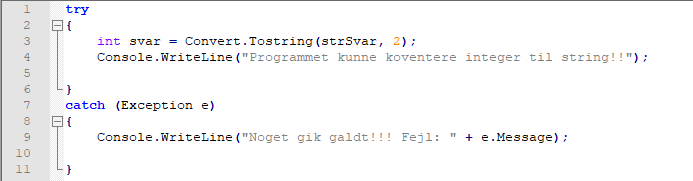


1. I dette tilfælde fik brugeren de første 3 rigtige, så han vinder!



# Konklusion

Jeg synes personligt at det var en lille smule kompliceret programmeringsprojekt i min mening. Det tog mig lidt under en time at skrive programmet, og det er nemmere for mig at skrive programmet før jeg laver et flowchart. Jeg har personligt lavet 3-4 forskellige versioner af programmet der mødete kravsspecifikationerne til punkt. Samlet set synes jeg det var et sjovt og intresandt forløb. Programmet fungere som det skal, og jeg har lavet en anden version hvor jeg har implemeteret kode der fixer problemer i programmet. De fejl jeg beskrev i ”problemer og fejl” længere op i opgaven. Så i næste version ville jeg bruge en try & catch model til at tjekke om vi overhoved kan konventere inddataen, ellers skriver den en fejl. En måde på at tjekke om brugeren overhoved har skrevet noget som svar.



# Vedhæftede Billag

Flowchart – ByteBuster.pdf

C# koden – program.cs

Opbygning af koden (2 sider) - Opbygning af koden.docx