

# Review af Alis Danske tal

## Overordnet

Link til github repo: <https://github.com/AliHMohammad/c-dansketal/tree/main>

Commit hash: a2d95dd

## Umiddelbare indtryk

Virker koden overskuelig, kompleks, rodet?

- Opgaven taget i betragtning, så er det en meget overskuelig løsning

Er der meget kode, eller er det kort og kompakt?

- 122 linjers kode. Vældig kompakt

Er der brugt mange funktioner, eller er det hele holdt i en?

- Én hjælpefunktion. Ellers holdes den primære funktionalitet til én funktion

Ser der ud til at være mange eller få gentagelser (gentagne mønstre)?

- Mønstrene går igen for de forskellige "talstørrelser". Men ikke mange unødige gentagelser af den grund

Giver det en følelse af at der har været for meget AI-hjælp inde over?

- Nej :)

## Funktionalitet

Virker programmet?

- Langt hen ad vejen, ja (mit eget går også infinite loop hvis man giver den et bogstav)

Skriver det korrekt?

- Det får tilsyneladende alle tallene med, men formateringen er ikke helt på

plads

**Kan det håndtere positive tal fra 0 til 2147483647?**

- Kan ikke håndtere milliarder

**Kan det håndtere negative tal fra -1 til -2147483648?**

- Samme som ovenfor, men ellers fint

**Er der ufundne fejl? Crashes eller tomme/forkerte strenge?**

- Alt andet end tal dræber programmet. Men nu er det også scanf. Ellers er de eneste fejl jeg kunne finde i form af formatteringen (som også er klart den sværste del, hvis du spørger mig). Eks: 1100 = tusind hundred

## Udvidelser

**Kan det konfigureres til at skrive "og" eller ikke?**

- Hvis man ændrer i source koden. Men den opfører sig som sagt ikke helt korrekt med alle formateringsmulighederne

**Kan det konfigureres til at skrive flertalsformer eller ej?**

- Samme som ovenfor

**Kan det konfigureres til at undlade "et" foran hundrede eller tusinde?**

- Samme som ovenfor (og der skrives "en" i stedet for "et" tusind/hundred)

**Kan det også håndtere kommatal?**

- Nej, crasher programmet

## Kode-struktur

**Er der brugt loops, switch-cases eller rekursion?**

- Rekursion brugt rimelig succesfuldt

**Hvordan er det håndteret at koden til hundreder fx både kan bruges ved hundrede-tusinde, hundrede alene, og som en underdel af tusinder?**

- Også gennem rekursion! Kodeeksempel:

```
void convert_to_danish(int number, char* buffer, struct NumberFormat format):
```

```
// andet kode
```

```
int thousand = number / 1000;
```

```
int rem = number % 1000;
```

```
convert_to_danish(thousand, buffer, format);
```

```
// andet kode
```

**Er der gentaget kode der kunne undgås med bedre brug af loops, hjælpefunktioner eller rekursion?**

- Der er ikke plads til meget spildkode på det korte program, så ikke rigtig. Måske et par linjer kunne spares ved at generalisere noget formartering yderligere.

## Streng-konkatanering

**Hvordan opbygges de lange resultat-streng?**

- strcat() fra <string.h> hele vejen.

## Særligt gode detaljer

- Elegant løsning, med god brug af rekursion. Overskuelig kode

## Særligt dårlige detaljer

- Formateringen halter på en del steder, men det er også det eneste.