Kryptografi

Læringsmål:

- Strenger
- Char
- Betingelser
- For-løkker

Pensum:

- 3.3 Input and Output
- 3.7 User-Defined Functions That Return a Single Value
- 4.1 The if statement
- 4.2 The if-else statement
- 4.3 Nested if-else statements
- 5.1 The for loop
- 7.2 Operations on Strings
- 7.3 The "is" Functions for Strings

Først litt historie:

En gang for lenge siden regjerte Julius Cæsar over Romerriket. Han var en svært suksessfull militær general og som et ledd i hans militære strategi skal han visstnok ha benyttet seg av det vi idag kaller for et Cæsarschiffer. Dette er en veldig enkel form for kryptering der hver bokstav i klarteksten erstattes med en annen bokstav et gitt antall steg lenger ut i alfabetet. For eksempel hvis vi skal skrive bokstaven **A** i cæsarchiffer og bruker en såkalt steglengde eller "nøkkel" på 3, vil vi få ut **D**.

Et eksempel på det engelske alfabetet og dets cæsarchiffer med nøkkel på **3**:

```
Klartekst: ABCDEFGHIJKLMNOPQRSTUVWXYZ
Chiffertekst: DEFGHIJKLMNOPQRSTUVWXYZABC
```

Som du kan se så hopper man tilbake til start når man kommer til slutten av alfabetet, slik at neste bokstav etter Z blir A etc.

a)

Lag funksjonen **`caesar(word, key)`**, som tar inn en bokstav **`word`** og forskyver hver bokstav (som er av typen **`char`**) i **`word`** med **`key`** antall tegn i alfabetet.
Bokstavene skal forskyves til høyre dersom step er positiv og til venstre dersom den er negativ.
Chifferet skal kun kjøres på bokstaver i det engelske alfabetet, dvs. **`a-z`**. Du kan anta at det kun tas inn små bokstaver.

Eksempel på kjøring

```
>> caesar('terningen', 3)
ans =

    'whuqlqjhq'
```

b)

Utvid nå funksjonen du lagde i deloppgave **a)** til å kunne håndtere setninger (hvis den ikke allerede gjør det).

Dersom funksjonen din støter på andre tegn enn **`a-z`**, slik som **`,.-;:`** og mellomrom osv. skal disse bare ignoreres. Her kan **`continue`** komme til nytte.

Lag deretter en funksjon **`plaintext(word, key)`** som tar inn et cæsarchiffer og dekrypterer dette. Dette er med andre ord en slags omvendt funksjon av `caesar()`.

Finner du en måte å gjøre dette på, uten å skrive så mye ny kode?

Eksempel på kjøring

```
>> caesar('terningen er kastet.', 3)
ans =

    'whuqlqjhq hu ndvwhw.'

>> plaintext('whuqlqjhq hu ndvwhw.', 3)
ans =

 'terningen er kastet.'
```

c)

Lag funksjonen **`safeTalk()`**. Den skal gi brukeren valget mellom å kryptere eller dekryptere et cæsarchiffer. Deretter skal den be brukeren om en tekstreng og antall steg. Den skal returnere den krypterte eller dekrypterte strengen.

Eksempel på kjøring

```
>> safeTalk
Vil du kryptere(k) eller dekryptere(d):
k

Hva vil du kryptere:
storebror ser deg

Hva er nøkkelen til chifferet?:
-6
ans =

            'mnilyvlil myl xya'
```