Hemmelege kodar

Skrevet av: Omsett frå Code Club UK (//codeclub.org.uk)

Oversatt av: Stein Olav Romslo

Kurs: Python
Tema: Tekstbasert, Kryptografi
Fag: Programmering, Samfunnsfag
Klassetrinn: 5.-7. klasse, 8.-10. klasse

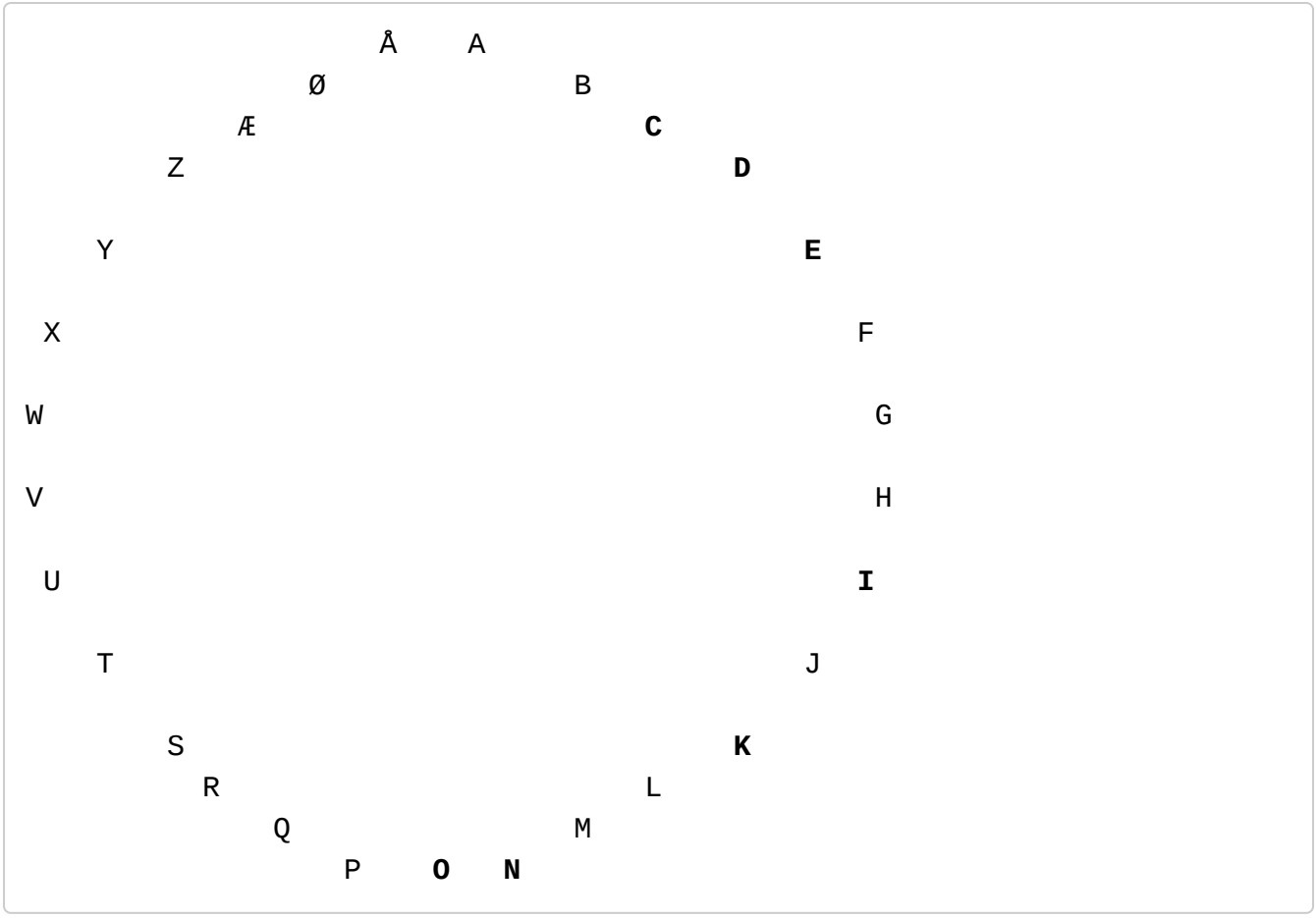
Introduksjon

Legg bort skjelpaddene dine, i dag skal me lære korleis me kan sende hemmelege beskjedar!

Kodeklubb-koden

Eit *chiffer* er eit system for å gjere om vanleg tekst til kode som ikkje andre skal kunne lese. Me skal bruke eit av dei eldste og mest berømte chiffera, nemleg Cæsar-chifferet eller Cæsar si kode - oppattkalla etter Gaius Julius Cæsar som sannsynlegvis brukte det til å sende hemmelege beskjedar. Det er nok ikkje den beste måten å hindre andre i å lese beskjedane dine, men det kjem me tilbake til. Det finst ferdige modular til Python du kan bruke viss du vil lage ein kode som skal vere vanskeleg å knekke, men no skal me prøve å lage Cæsar-chifferet sjølv.

Start med å teikne alle bokstavane i ein sirkel.



For å lage ein hemmeleg bokstav frå ein vanleg bokstav treng me eit tal me kan bruke som hemmeleg nøkkel. Her brukar me talet 3.

```
A + 3 = D       T + 3 = W       Å + 3 = C
```

Me startar med A og tel framover 3 bokstavar: B, C, D. Så bokstaven A blir bokstaven D i koden vår. For å dekode gjer me det same, men baklengs. Me startar med D og tel 3 bakover for å få A.

Steg 1: Alfabetet

Her kan du få trøbbel med norske bokstavar viss du ikkje har Python 3. Du har Python 2 viss det står 2.6 eller 2.7 i IDLE. I så fall må du leggje ein `u` framfor tekst som er inni `""`. Til dømes blir alfabetet under `u"abcdefghijklmnopqrstuvwxyzæøå"`. Når me legg til `u`-en seier me at teksten er av typen *Unicode* som støttar alle norske bokstavar.

Sjekkliste

- [ ] Fyrst må me lære Python alfabetet. Åpne IDLE og lag ei ny fil med koden under:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

print(len(alphabet))
```

- [ ] Når du køyrer dette skriptet skal det skrive ut 29. Pass på at du har med alle bokstavane, elles kjem ikkje den hemmelege koden din til å fungere.

  Viss du har fått det til kan me starte med å kode ein bokstav.

Steg 2: Kode ein bokstav

Sjekkliste

- [ ] Akkurat som me gjorde med sirkelen over kan me finne posisjonen til ein bokstav ved å telje framover, og så bruke bokstaven me endar opp med.

  Skriv inn koden under og køyr den:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

letter = "a"
secret = 3

pos = alphabet.find(letter)

newpos = (pos + secret)

if newpos >= 29:
    newpos = newpos - 29

secretletter = alphabet[newpos]

print(secretletter)
```

Me slår opp kor `a` er i alfabetet og legg til det hemmelege talet vårt for å telje framover. Me sjekkar om me har gått rundt, og viss me har det må me gå ein heil runde tilbake ved å trekkje frå 29. Dette er litt som med gradene på ein sirkel, viss me trekk frå 360 ender me akkurat der me sto i utgangspunktet. Så slå me opp i alfabetet att for å sjå kva den hemmelege bokstaven er.

- [ ] Køyr koden og sjå kva som skjer.

- [ ] La oss sjå på koden att, men me gjer det sakte.

  Du treng ikkje skrive dette! Alt som står bak `#` bryr Python seg vanlegvis ikkje om, det er berre kommentarar til menneske som skal lese koden.

```
# alphabet er namnet på teksten frå a til å
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

# Den hemmelege bokstaven (letter) og det hemmelege talet
# (secret) me brukar for å kode det
letter = "a"
secret = 3

# Finn posisjonen til bokstaven. Python vil gi oss eit
# tal frå 0 til 28 (python tel frå 0)
pos = alphabet.find(letter)

# Gå like langt framover som det hemmelege talet seier
newpos = (pos + secret)

# Viss me har telt for langt må me gå ein runde attende
# for å få eit tal mellom 0 og 28
if newpos >= 29:
    newpos = newpos - 29

# Slå opp denne posisjonen for å sjå kva bokstav
# i alfabetet som står der
secretletter = alphabet[newpos]

# Skriv ut denne bokstaven på skjermen
print(secretletter)
```

  Det er mange Pyhton-ting som skjer her, men ikkje bli skremt viss du ikkje forstår alt med ein gong. Mykje av dette er akkurat som i Scratch. Linja `if newpos >= 29` er berre ei `if`-setning, ein kontroll som berre køyrer koden under viss det som står etter `if` er sant. Ei `if`-setning brukar ein innrykksblokk, akkurat som `for` og `def` som me har sett tidlegare.

Nå som me kan kode ein bokstav, kva med å dekode ein?

Steg 3: Finne att bokstavane

Akkurat som i koden frå den førre oppgåva skal me finne posisjonen til bokstaven, men denne gongen skal me gå bakover i alfabetet for å dekode.

Sjekkliste

- [ ] Prøv å skrive denne koden og køyr den:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

secret = 17
secretletter = "r"

pos = alphabet.find(secretletter)

newpos = pos - secret

if newpos < 0:
    newpos = newpos + 29

letter = alphabet[newpos]

print(letter)
```

Steg 4: Byggje funksjoner

La oss ta koden som lagar og les Cæsar-kodar og gjere den om til to *funksjonar*. Gi den eine funksjonen namnet `encode` og den andre funksjonen namnet `decode`. **Tips:** Viss du ikkje har høyrt om funksjonar kan du lese meir om dei i Skjelpaddeskulen (../skilpaddeskolen/skilpaddeskolen_nn.html).

For å få ein funksjon til å sende attende ein verdi brukar me `return`. Dette gjer at me kan lagre resultatet frå funksjonen i ein variabel, og så bruke variabelen.

Sjekkliste

- [ ] Lag ei fil som ser slik ut:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

def encode(letter, secret):
    pos = alphabet.find(letter)

    newpos = (pos + secret)

    if newpos >= 29:
        newpos = newpos - 29

    return alphabet[newpos]


def decode(letter, secret):
    pos = alphabet.find(letter)

    newpos = (pos - secret)

    if newpos < 0:
        newpos = newpos + 29

    return alphabet[newpos]

print(encode("a", 17))
print(decode("r", 17))
```

Hugs at du kan bruke tabulatortasten til venstre på tastaturet (rett over caps lock) i IDLE for å få innrykk. Du kan merke delar av koden og trykke tabulatortasten for å rykke inn alt på ein gong.

- [ ] Prøv å kode og dekode nokre bokstavar!

Steg 5: Send eit hemmeleg ord eller to, og finn dei att

No har me nokre funksjonar, la oss bruke dei til å kode ord. Me kjem til å gå gjennom kvar bokstav i ordet, og kode det viss det finst i alfabetet (me hoppar over teikn som punktum og mellomrom).

Sjekkliste

- [ ] Under dei nye funksjonane frå førre oppgåve kan du skrive inn koden under (med andre ord: behald det du gjorde i steg 4, og legg til koden under).

```
secret = 17
message = "hello world"

output = ""

for character in message:
    if character in alphabet:
        output = output +  encode(character, secret)
    else:
        output = output + character


print(output)

secret = 17
message = "yvåac kcfåu"
output = ""

for character in message:
    if character in alphabet:
        output = output + decode(character, secret)
    else:
        output = output + character

print(output)
```

- [ ] Køyr programmet og sjå kva som skjer.

  Den fyrste delen av koden burde skrive ut `"yvåac kcfåu"`, som er den hemmelege versjonen av `"hello world"`. Den andre delen dekodar det att og skriv ut `"hello world"`.

Steg 6: Dekoding av nokre hemmelege beskjedar

Her er nokre hemmelege beskjedar. Prøv å dekode dei!

- [ ] `daczj ym cgyzcdmwwzf?`, hemmelegheita er 21.

- `æxkxånwn næ bnwwnwn mrwn`, hemmelegheita er 9.

Prøv å sende nokre beskjedar til venene dine! Kva med å lage eit Python-program som prøver seg på alle moglege hemmelege tal og prøver å knekke kodar sjølv om du ikkje kan det hemmelege talet?

Lisens: Code Club World Limited Terms of Service (https://github.com/CodeClub/scratch-curriculum/blob/master/LICENSE.md)