Hemmelige koder

Skrevet av: Oversatt fra Code Club UK (//codeclub.org.uk)

Oversatt av: Bjørn Einar Bjartnes

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

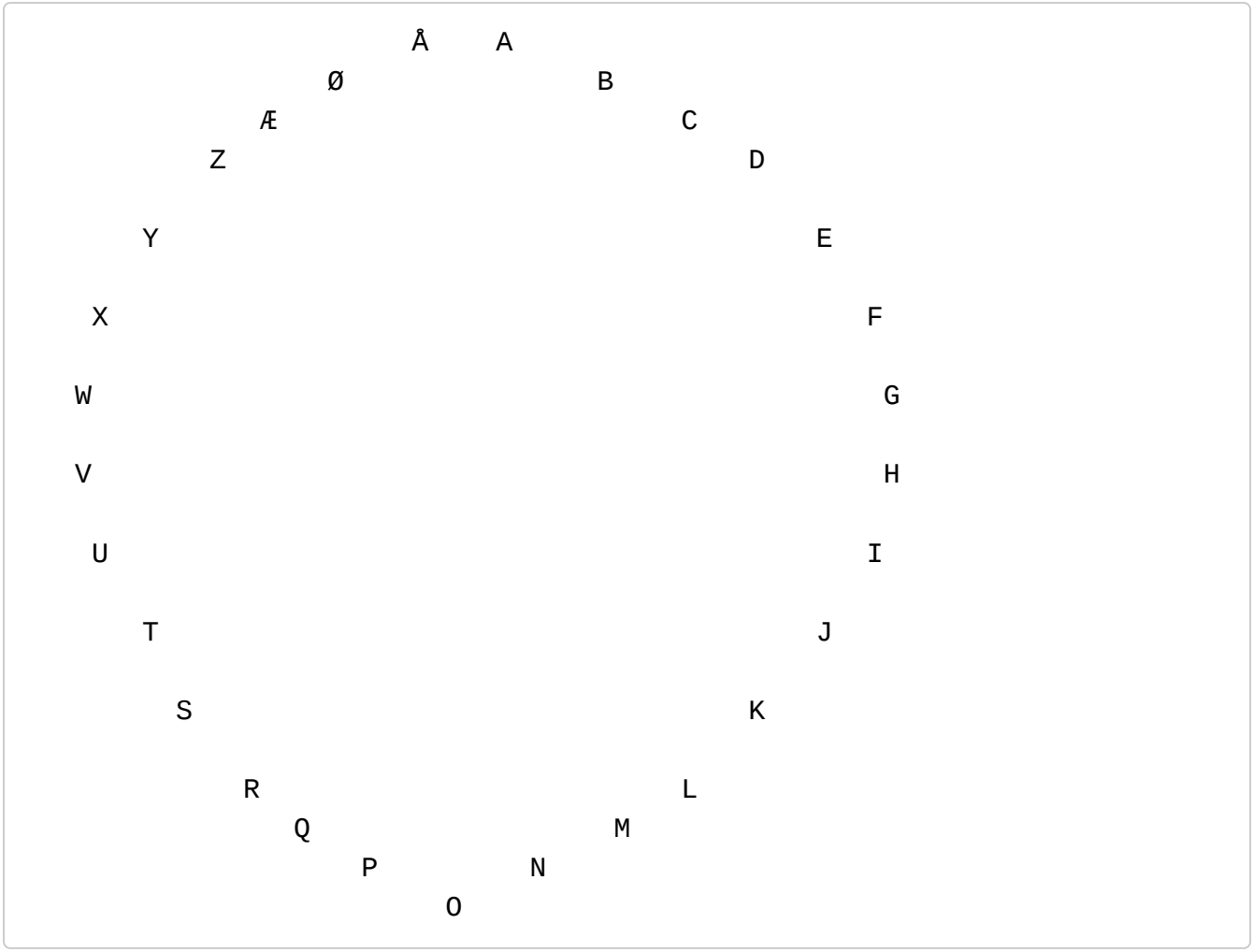
Introduksjon

I dag skal vi lære hvordan vi kan sende hemmelige beskjeder!

Kodeklubb-koden

Et chiffer er et system for å gjøre om vanlig tekst til kode som ikke andre skal kunne lese. Vi skal bruke et av de eldste og mest berømte chifferene, Cæsar-chifferet eller Cæsars kode - oppkalt etter Gaius Julius Cæsar som sannsynligvis brukte det til å sende hemmelige beskjeder. Det er neppe den beste måten å hindre andre i å lese beskjedene dine, men det kommer vi tilbake til. Det finnes ferdige moduler til Python du kan bruke hvis du vil lage noe som skal være vanskelig å knekke, men nå skal vi forsøke å lage Cæsar-chifferet selv.

Start med å tegne alle bokstavene i en sirkel.



For å lage en hemmelig bokstav fra en vanlig bokstav, trenger vi et tall vi kan bruke som hemmelig nøkkel. Jeg liker tallet 3, så vi bruker det.

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

Vi begynner med A og teller fremover 3 bokstaver: B, C, D. Så bokstaven A blir til bokstaven D. For å dekode gjør vi det samme, men baklengs. Vi begynner med D og teller bakover for å få A.

Steg 1: Alfabetet

Sjekkliste

- [ ] Først må vi lære python alfabetet. Åpne IDLE og lag en ny fil med koden under:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

print(len(alphabet))
```

- [ ] Når du kjører dette programmet skal det skrive ut 29. Pass på at du har med alle bokstavene, ellers kommer ikke den hemmelige koden din til å virke.

  Hvis du er fornøyd med alfabetet ditt kan vi begynne å kode en bokstav.

Steg 2: Kode en bokstav

Sjekkliste

- [ ] Akkurat som vi gjorde med hjulet ovenfor kan vi finne posisjonen til en bokstav ved å telle forover, og så bruke bokstaven vi ender opp med.

  Skriv inn koden under og kjør den:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

letter = "a"
key = 3

pos = alphabet.find(letter)

newpos = (pos + key)

if newpos >= 29:
    newpos = newpos - 29

secretletter = alphabet[newpos]

print(secretletter)
```

  Vi slår opp hvor "a" er i alfabetet og legger til den hemmelige nøkkelen vår for å telle fremover. Vi sjekker om vi har gått rundt, hvis vi har det må vi gå en hel runde tilbake igjen ved å trekke fra 29. Så slår vi opp i alfabetet igjen for å se hvilken hemmelige bokstav vi fikk.

- [ ] Kjør koden og se hva som skjer.

- [ ] La oss ta en titt på koden igjen, men vi tar det sakte.

```
# alphabet er navnet på teksten fra a til å
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

# Den hemmelige bokstaven (letter) og det hemmelige tallet
# (key) vi bruker for å kode det
letter = "a"
key = 3

# Finn posisjonen til bokstaven. Python vil gi oss et
# tall fra 0 til 28 (python teller fra 0)
pos = alphabet.find(letter)

# Gå like langt fremover som det hemmelige tallet sier
newpos = (pos + key)

# Hvis vi har telt for langt, må vi gå en runde tilbake
# for å få et tall mellom 0 og 28
if newpos >= 29:
    newpos = newpos - 29

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

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

Nå som vi kan kode en bokstav, hva med å dekode en?

Steg 3: Finne tilbake bokstavene

Akkurat som i koden fra den forrige oppgaven skal vi finne posisjonen til bokstaven, men denne gangen skal vi gå bakover i alfabetet for å dekode.

Sjekkliste

- [ ] Forsøk å skriv inn denne koden og kjør den:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

key = 17
secretletter = "r"

pos = alphabet.find(secretletter)

newpos = pos - key

if newpos < 0:
    newpos = newpos + 29

letter = alphabet[newpos]

print(letter)
```

Steg 4: Bygge funksjoner

La oss ta koden som lager og leser Cæsar-koder og gjøre den om til to *funksjoner*. Gi den ene funksjonen navnet `encode` og den andre funksjonen navnet `decode`. **Tips:** Dersom du aldri har hørt om funksjoner, kan du lese mer om de i Skilpaddeskolen (../skilpaddeskolen/skilpaddeskolen.html).

For å få en funksjon til å sende tilbake en verdi bruker vi `return`. Dette gjør at vi kan lagre funksjonens resultat til en variabel og deretter bruke variabelen.

Sjekkliste

- [ ] Lag en fil som ser slik ut:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøå"

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

    newpos = (pos + key)

    if newpos >= 29:
        newpos = newpos - 29

    return alphabet[newpos]


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

    newpos = (pos - key)

    if newpos < 0:
        newpos = newpos + 29

    return alphabet[newpos]

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

- [ ] Prøv å kode og dekode noen bokstaver!

Steg 5: Send et hemmelig ord eller to, og finn dem tilbake igjen

Nå har vi noen funksjoner, la oss bruke dem til å kode ord. Vi kommer til å gå igjennom hver bokstav i ordet og kode det hvis det finnes i alfabetet (vi hopper over tegn som punktum og mellomrom).

Sjekkliste

- [ ] Under de nye funksjonene fra forrige oppgave kan du skrive inn koden under (med andre ord: behold det du gjorde i oppgave 4, og legg til følgende kode).

```
key = 17
message = "hello world"

output = ""

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


print(output)

key = 17
message = "yvååc kcfåu"
output = ""

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

print(output)
```

- [ ] Kjør programmet og se hva som skjer.

  Den første delen av koden burde skrive ut "yvååc kcfåu", som er den hemmelige versjonen av "hello world". Den andre delen dekoder det igjen og skriver ut "hello world"

Steg 6: Bygge flere funksjoner

På samme måtte som vi skrev funksjoner for å kode og dekode bokstaver, så ønsker vi å lage funksjoner for å kryptere og dekryptere hele meldinger.

- [ ] Skriv en funksjon `encrypt` som tar som input `message` og `key`, og returnerer den krypterte meldingen under denne nøkkelen.

- [ ] Skriv en funksjon `decrypt` som tar som input `secretmessage` og `key`, og returnerer den dekrypterte meldingen under denne nøkkelen.

Steg 7: Utvide alfabetet og forbedre koden

Vi ønsker å kunne kryptere ulike tegn, ikke bare små bokstaver. Da må vi gjøre programmet vårt litt mer fleksibelt, ettersom vi har sagt at koden vår bare fungerer skikkelig dersom vi har 29 tegn i alfabetet. Vi ønsker i første gang å legge til store bokstaver, men du kan også legge til spesialtegn som `?` eller `!`.

- [ ] Skriv om de første linjene i koden din. Først utvider vi alfabetet med å legge til store bokstaver, og så legger vi til en ny variabel `l`:

```
alphabet = "abcdefghijklmnopqrstuvwxyzæøåABCDEFGHIJKLMNOPQRSTUVWXYZÆØÅ"
l = len(alphabet)
```

Nå har vi lagt til flere bokstaver, og lagret lengden av alfabetet i variabelen `l`.

- [ ] Bytt ut tallet `29` med variablen `l` alle steder i programmet ditt.

- [ ] Til slutt vil vi ende litt på koden hvor vi trekker fra lengden på alfabetet. Dersom vi skriver `a % b` i koden, så betyr dette at vi får resten av `a` etter at vi har delt på `b`. For eksempel vil `7 % 5` gi oss `2`, fordi `7` er `2` større enn `5`. Gå til funksjonen `encode` og bytt om koden:

```
newpos = (pos + key)
```

med følgende:

```
newpos = (pos + key) % l
```

Nå kan vi **fjerne** kodebiten

```
if newpos >= l:
      newpos = newpos - l
```

fordi linjen vi nettopp endret gjør akkurat det samme. Dersom `newpos` er større enn `l`, så blir den automatisk juster til et tall som er mindre enn `l`.

- [ ] Endre funksjonen `decode` på samme måten som vi endret `endcode`.

Steg 8: Dekryptering av noen hemmelige beskjeder

Her er noen hemmelige beskjeder, forsøk å dekode dem!

- [ ] Kryptert melding: `daczj ym cgyzcdmwwzf?`, nøkkel: `21`.

- [ ] Kryptert melding: `yvivælul ly åluulul kpul`, nøkkel: `7`.

- [ ] Kryptert melding: `Æxø, åxz IøJJxH Ez AEwxH!`. Den hemmelige meldingen starter med `Hei`. Hva er nøkkelen? Hva er den hemmelige meldingen?

- [ ] Kryptert melding: `CÆÅ Åvk iv dhZÆdenXXÆg ØhkZb cÆÅ ebdÆk v dhZÆ`. Den hemmelige meldingen inneholder ordet `kodeklubben`. Hva er nøkkelen? Hva er den hemmelige meldingen?

- [ ] Kryptert melding:

  ```
  qMOHPIZHQSSMHØQLHØQTHgHORfZMHTMSÆMZHNWZLQHRMOHMZHWXXØIØØHUMLHgHSWLM
  ```

  Den hemmelige meldingen inneholder en del vanlige norske ord. Hva er nøkkelen? Hva er den hemmelige meldingen?

  PS: Her har vi lagt til mellomrom på slutten av alfabetet vårt!

Prøv å sende noen beskjeder til vennene dine!