◆ Kryptonøtt

Skrevet av: Arve Seljebu

Kurs: Python
Tema: Tekstbasert
Fag: Programmering
Klassetrinn: 8.-10. klasse, Videregående skole

Introduksjon

Kryptering har lenge vært i bruk i kommunikasjon. Faktisk brukte de det for nesten 4000 år siden!! I tillegg er det artig å sende hemmelige meldinger :-) Før du begynner på denne oppgaven, anbefales det at du har gjort [Hemmelige koder] først.

Denne oppgaven er en nøtt. Det vil si at du skal finne ut av det meste selv. Sitter du helt fast må du gjerne spørre en CodeMaster.

Kryptering med vigenere-metoden

Vigenere er litt smartere enn krypteringen i [Hemmelige koder], men den er ikke så annerledes. I stegene under skal du prøve å forstå vigenere-koden. Det er viktig at du forstår denne koden, ettersom du skal lage nesten lik kode selv.

Python 2

Denne koden fungerer best med python 3. Dersom du har python 2, må du legge en `u` foran alle strenger. Altså `'asdf'` må skrives slik som dette: `u'asdf'`.

Lag kommentarer med forklaring

- [ ] Les koden under.

- [ ] Hva er forskjellig fra [Hemmelige koder]?

- [ ] Hva gjør `alphabet.find` ?
- [ ] Hva betyr det at `alphabet.find` gir `-1` som svar?
- [ ] Legg til kommentarer med `#` over/bak hver linjene med din forklaring.

```
2014"""

alphabet = 'ABCDEFGHIJKLMNOPQRSTUVWXYZÆØÅabcdefghijklmnopqrstuvwxyzæøå
.,?-_;:+1234567890"'

def vigenere_encode(msg, key):
    """Function that encodes a string with Vigenere cipher. The encrypted
       string is returned. """ secret = '' key_length = len(key) alphabet_leng
th
    = len(alphabet)

    for i, char in enumerate(msg):
        msgInt = alphabet.find(char) encInt = alphabet.find(key[i % key_lengt
h])

        if msgInt == -1 or encInt == -1:
            return ''

        encoded = (msgInt + encInt) % alphabet_length secret +=
        alphabet[encoded]

    return secret

message = 'My first computer program was a song called Popcorn written in
QBasic. The second computer program I made was a bot made for IRC.' keyword =
'source'

encrypted = vigenere_encode(message, keyword) print(encrypted) ```

## Hint {.protip}

Du kan bruke kommandoen `help('funksjonsnavn')` i python-terminalen for lese
manualen. Prøv disse:

- `help('def')`

- `help('len')`

- `help('vigenere_encode')`


# Dekryptering {.activity}

Vi skal nå se på hvordan vi kan dekryptere meldinger. Etterhvert vil vi til og
med kunne lese hemmelige meldinger uten å kjenne den hemmelige nøkkelen på
forhånd.

## Lag vigenere_decode {.check}
```

Lag en funksjon som gjør det motsatte av den over (altså dekrypterer). Koden skal se nesten helt lik ut som over.

- [ ] Funksjonen skal ta inn to parametre: en kodet tekst og en nøkkel.

- [ ] Den skal dekryptere den kodede teksten med nøkkelen.

- [ ] Og returnere den dekrypterte teksten.

- [ ] Test at funksjonen fungerer og prøv med dine egne strenger og krypteringsnøkler.

- [ ] Kanskje du kan dele nøkkelen og sende den krypterte teksten til en venn?

Cracking {.check}

- [ ] Du skal nå prøve å knekke en kodet streng. Dette er vanskelig, så du må lage en plan først. Strengen er:

`q0Ø:;AI"E47FRBQNBG4WNB8B4LQN8ERKC88U8GEN?T6LaNBG4GØ""N6K086HB"Ø8CRHW"+LS79Ø""N29QCLN5WNEBS8GENBG4FØ47a`

Hint {.protip}

- Nøkkelen er seks små bokstaver.

- Språket i setningen er engelsk.

- Finn en metode å sjekke om den dekrypterte strengen er korrekt. For eksempel kan du tenke på hvor mange mellomrom den burde inneholde?

- For å generere mulige nøkler kan du bruke `itertools.product()`, prøv for eksempel å se hva du får om du looper over `itertools.product('abcd', repeat=2)`.

Bruk en ordbok {.check}

Sålenge vi har brukt engelske ord som nøkler er det mye raskere å knekke krypteringen med en ordbok. En ordbok finner du på alle Linux/Mac/Unix-maskiner
under **/usr/share/dict**. Bruker du Windows, kan du laste ned en slik fil fra internett. Søk på *large english vocabulary word lists*.

- [ ] Disse filene inneholder alle ord som finnes i en engelsk ordbok, separert
      med linjeskift. Finn ut hvordan du kan laste inn ordene fra filen (pass på

```
      at du fjerner linjeskiftene) og bruk dem til å dekryptere en ny streng:

`t-JO:BK0aM,:CQ+ÆAGW?FJGB0KVCGMQ6SQN"GAIDL-PÅ7954E:7Jr,IÆoCF0M"CQdØVlHD53CÅ;IA
2DMG5ØHDØVåL:JQØ439LRBBVEMTBÆ6CF0M"CQNAG8G1V6LÅ8FF4Z`

- [ ] Bruk metodene du laget i oppgaven over for å detektere om vi har funnet
      riktig nøkkel. Dersom du kjører scriptet ditt med kommandoen `time pytho
n3
      vigenere.py` kan du se hvor lang tid den bruker.

## Premie {.flag}

Dersom du klarer denne nøtten, spanderer jeg gjerne en sjokolade på deg dersom
du deler koden din. Send en epost til arve@seljebu.no :-)

[Hemmelige koder]: ../hemmelige_koder/hemmelige_koder.html
```

Lisens: CC BY-SA 4.0 (http://creativecommons.org/licenses/by-sa/4.0/deed)