

# IN1140, H2021 – Gruppetime

## Regulære uttrykk

Regulære uttrykk brukes for å karakterisere tekstsekvenser innenfor en rekke typer tekstprosessering og informasjonsekstraksjon. Vi skal her jobbe med å skrive regulære uttrykk, samt å implementere dem i Python. Vi anbefaler også å jobbe med Trix-oppgavene rundt samme tema på egenhånd.

### 1 Anvend et regulært uttrykk

1. Hvilket av disse navnene gjenkjennes med følgende regulære uttrykk:

```
/[A-Z][a-z]+\.\s[A-Za-z]+\s[A-Za-z]+/
```

Velg ett alternativ:

- (a) Pres. Barack Obama
  - (b) President Obama
  - (c) Pres. B. Obama
  - (d) President B. Obama
2. Hvilken av de følgende epost-adressene kan ikke gjenkjennes med det følgende regulære uttrykket:

```
[aA-zZ0-9\._%+-]+@[aA-zZ0-9\.-]+\.[aA-zZ]{1,2}
```

Velg ett alternativ:

- (a) Ola.Norman@epost.no
- (b) OlaNorman@meg.co
- (c) Ola.Norman@epost.com
- (d) olanorman@meg.epost.no

### 2 Skriv egne regulære uttrykk

1. Skriv et regulært uttrykk som gjenkjenner datoer av typen “fredag 15/4”, “mandag 6/12” eller “torsdag 30/5”. Uttrykket ditt skal ikke gjenkjenne umulige datoer av typen “mandag 36/12” eller “fredag 25/14”, men du trenger ikke å ta høyde for riktig antall dager i hver måned (slik som eksempelvis å utelukke 30/2).

Bruk Python for å løse oppgaven og skriv kode som gjenkjenner mulige datoer fra følgende liste:

```
liste = ['mandag 6/12', 'tirsdag 30/5', 'onsdag 36/12',  
'torsdag 25/14', 'januar 01/01', 'lørdag 1/5', 'søndag 07/07']
```

2. Skriv et regulært uttrykk som gjenkjenner strenger av 0 og 1 med minst to etterfølgende 0'ere. Den skal f.eks. gjenkjenne strengene 010001, 00, 10101001, 01011000 men ikke strengene 01, 0101, 01111101111.
3. Skriv et regulært uttrykk som gjenkjenner følgende måte å lese klokkeslett på:
  - 6 o'clock
  - quarter past 5
  - half past 9
  - quarter to 2
  - 3 to 7
  - 10 past 11
4. Skriv et regulært uttrykk som gjenkjenner følgende strenger:
  - grønn, grønnere, grønnest, god, bedre, best, smart, smartere, smartest