Obligatorisk innlevering 1b - Regulære uttrykk & tokenisering med regulære uttrykk i Python

Høst 2025

Husk å lese gjennom tilbakemeldingen din på forrige innlevering i Devilry før du leverer denne.

Innleveringsfrist: mandag 6. oktober kl. 23.59

Innlevering av oppgaven skjer i Devilry. Det er viktig at du setter deg inn i retningslinjene for obligatoriske oppgaver. Besvar oppgavene i én fil som angir brukernavnet ditt slik:

`oblig1b_brukernavn.py`

Du skal også levere inn filen `in01.txt` som du fikk utlevert med forrige innlevering. Den skal brukes i denne innleveringen også. Ikke pakk filene inn i en ZIP-mappe. Besvarelsen må kunne kjøres med Python i terminalen, slik som tidligere.

Poeng

En perfekt løsning av denne innleveringen er verdt 100 poeng. Koden må kunne kjøres og må inneholde kommentarer som forklarer hva koden gjør. Løsningen på oppgave 1, 2, 3 og 4.1 skal skrives som kommentarer i Python-filen din. Du trenger kun å programmere i oppgave 4.2 og 4.3.

1 Regulære uttrykk (eksamen høsten 2018) (10 poeng)

Hvilket av alternativene kan ikke gjenkjennes med det følgende regulære uttrykket:

```
[1-9][0-9]*\s((cent(s)?)|(dollar(s)?\s+([1-9][0-9]*\scent(s)?)))
```

Velg ett alternativ:

1. `1 dollar 35 cents`
2. `35 dollars`
3. `99 cents`
4. `99 dollars 1 cent`

2 Regulært uttrykk for verb (eksamen høsten 2017) (20 poeng)

Skriv ett regulært uttrykk som gjenkjenner formene imperativ, infinitiv, presens og preteritum av følgende norske verb: *spise*, *kjøpe*, *tenke*, *rope*. Du skal skrive ett enkelt uttrykk som kun gjenkjenner verbene med formene spesifisert. Bruk tabellen som referanse.

Infinitiv	Imperativ	Presens	Preteritum
spise	spis	spiser	spiste
kjøpe	kjøp	kjøper	kjøpte
tenke	tenk	tenker	tenkte
rope	rop	roper	ropte

3 Regulært utrykk for filnavn (20 poeng)

Skriv ett regulært uttrykk som gjenkjenner filnavn som ender på:

- .gif
- .jpg
- .png
- .jpeg
- .pdf
- .rtf

Husk at selve navnet også må gjenkjennes (altså f.eks. `utgave_3.pdf` og `Møre_og_Romsdal.jpeg`), ikke bare endelsen alene. Du kan anta at navnet kun kan bestå av bokstaver, tall og enkel tegnsetting som understrek og bindestrek, og at det må bestå av minst ett tegn.

4 Tokenisering med regulære uttrykk (50 poeng)

I denne oppgaven skal vi jobbe med å forbedre tokeniseringen fra **innlevering 1a**, basert på feilanalysen fra oppgave 5 i innlevering 1a. Du skal her forsøke å rette noen av de feilene du fant under feilanalysen din.

1. Vi starter med å formulere et regulært uttrykk som definerer et gyldig ord. Her bør du bruke disjunksjon for å beskrive forskjellige typer ord. For eksempel, bør det regulære uttrykket ditt skille mellom bokstavsekvenser og tegnsetting, da et gyldig ord ikke inkluderer tegnsetting, og ulike typer tegnsetting skal skilles ut som egne token. Eksempelvis bør *"Løp!"* tokeniseres som `Løp` og `!` hver for seg. Kommenter kort hvordan uttrykket ditt fungerer.
2. Benytt deg av `re.findall()` for å hente ut ordene som gjenkjennes av uttrykket ditt, fra teksten i filen `in01.txt`. Denne metoden tar inn et regulært uttrykk og en streng, og returnerer en liste med alle treff. F.eks. vil `re.findall('[0-9]','in1140')` returnere listen `['1', '1', '4', '0']`. Bruk det regulære uttrykket du lagde i **oppgave 4.1**. Du kan definere det i koden din slik: `ru = r'DITT UTTRYKK HER'`
3. Skriv ut tokeniseringen til en fil slik som i **innlevering 1a**, og inspiser resultatet av den forbedrede tokeniseringen. Gjenstår det noen feil? Gi i så fall noen eksempler.