▲ Lærerveiledning - Hash-funksjoner

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

Om oppgaven

Denne oppgaven inngår i en serie om kryptografi, og handler om hvordan man kan beskytte en melding mot feil mens den er på vei fra A til B. Det anbefales å gå gjennom følgende oppgaver før man starter på denne:

- ○ Hemmelige koder (../hemmelige_koder/hemmelige_koder.html)

Oppgaven er ikke testet på hele målgruppen, så tilbakemeldinger på nivået og egnede trinn er velkomne.

✔ Oppgaven passer til:

Fag: Programmering, matematikk

Anbefalte trinn: 8. trinn--VG3

Tema: Kryptografi, primtall, IT-sikkerhet

Tidsbruk: Dobbeltime

Kompetansemål

- ☐ **Valgfag programmering**: Prinsipper som ligger til grunn for god programmeringspraksis inngår også i hovedområdet, deriblant forklaring og dokumentasjon av løsninger og programkode; vurdering og analyse av egen og andres programkode (Fra hovedområdene)
- ☐ **Valgfag programmering**: omgjøre problemer til konkrete delproblemer

- [ ] **Matematikk X**: gjøre rede for praktiske anvendelser av kongruensregning i kryptering og feilrettingskoder

Forslag til læringsmål

- [ ] Elevene forstår hvordan et sjekksiffer beskytter strekkoder mot små lesefeil
- [ ] Elevene forstår hvorfor og hvordan man kan beskytte en melding mot endring underveis

Forslag til vurderingskriterier

- [ ] Eleven oppnår middels måloppnåelse ved å fullføre oppgaven opp til siste punkt
- [ ] Eleven oppnår høy måloppnåelse ved å også fullføre siste punkt, og kunne sende og motta vilkårlige meldinger med tilhørende MAC

Forutsetninger og utstyr

- [ ] **Forutsetninger**: God kjennskap til Python, noe matematisk modenhet. Gjennomført tidligere oppgaver som beskrevet over.
- [ ] **Utstyr**: Datamaskin med Python installert

Fremgangsmåte

Vi har dessverre ikke noen konkrete tips, erfaringer eller utfordringer tilknyttet denne oppgaven enda.

På de laveste trinnene kan temaet kan virke matematisk krevende når en ser på det første gang. Derfor kan det kanskje være nyttig å først og fremst angripe det fra et programmeringsperspektiv, for koden i seg selv er ikke særlig komplisert. I neste omgang kan man da bruke det en har programmert for å forstå matematikken bedre.

Variasjoner

Eksterne ressurser

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