

Terminavslutning

Oppgave 0 - Utrekninger

a

Lag et program som ber brukeren om et tall og skriver tilbake kvadratroten av tallet.

b

Utvid programmet slik at dersom brukeren skriver inn et negativt tall, svarer programmet `Jeg beklager, jeg har ikke lært om imaginære tall ennå.`

Oppgave 1 - Produktsammenligningen

Lag en funksjon, kall den `prod_gt_n`, som tar inn tre tall, `a`, `b` og `n`, og returnerer `True` dersom absoluttverdien til produktet av `a` og `b` er større enn `n`.

Funksjonen skal bestå følgende tester:

a	b	n	prod_gt_n(a, b, n)
1	1	2	False
1	2	2	False
3	2	2	True
-1	-1	-1	True
5	5	24	True
3	-3	8	True
3	3	10	False

Oppgave 2 - «An odd function»

a

Hva gjør funksjonen `f` under?

```
def f(x):
    if x % 2 == 0:
        return True
    else:
        return False
```

b

Ivar mener at funksjonen kan skrives som følger:

```
def f(x):
    return x % 2 == 0
```

Har Ivar rett i det?

Oppgave 3 - Derivasjon av datafil

I fila `posisjonsmaalinger_til_test.txt` er det to kolonner med data. De fem første linjene i fila er vist under.

```
0.05;0.9893835
0.1;0.9921275
0.15;0.989212
0.2;0.98784
0.25;0.9876685
```

Legg fila i samme mappe som du har python-fila du skriver svaret til oppgaven i. Du kan åpne fila og få tilgang til dataene som vist under.

```
posisjonsfil = open("posisjonsmaalinger_til_test.txt", "r", encoding="utf-8-sig")
filtekst = posisjonsfil.read().splitlines()
posisjonsfil.close()
```

`filtekst` er nå en liste av strenger, hvor hver streng er en linje fra fila.

a

Opprett to lister, `tider` og `posisjoner`, disse skal vi bruke til å samle tider og posisjoner fra fila.

Bruk deretter en `for`-løkke til å gå gjennom `filtekst`. For hver linje, gjør følgende:

1. `split` linja ved semikolonet, `;` og lagre dette som en midlertidig liste
 2. `append` flyttallsverdien til det første elementet til lista `tider`
 3. `append` flyttallsverdien til det andre elementet til lista `posisjoner`
- (Kontroll: `sum(tider) == 16.25` og `sum(posisjoner) == 19.0922375`.)

b

Lag en funksjon `delta` som tar inn en liste `inn_liste` og returnerer en liste, `ut_liste`, med følgende egenskaper:

1. `len(ut_liste) == len(inn_liste) - 1`
2. `ut_liste[i] = inn_liste[i+1] - inn_liste[i]`

c

Bruk `delta` til å lage `delta_posisjon` og `delta_tid` som

```
delta_posisjon = delta(posisjoner)
delta_tid = delta(tid)
```

(Kontroll: `sum(delta_posisjon) == -0.8082795` og `sum(delta_tid) == 1.2`.)

d

Finn en liste `hastigheter` som er slik at `hastigheter[i] = delta_posisjon[i] / delta_tid[i]`.

(Kontroll: `sum(hastigheter) == -16.16559`.)

e - Kan løses uten å ha fullført de øvrige deloppgavene

Finn gjennomsnittsfarten til gjenstanden som total avstand delt på total tid.