3. september, 2015

MAT-INF 1100: Obligatorisk oppgave 1

Innleveringsfrist: 17/9-2015, kl. 14:30

Informasjon

Den skriftlige besvarelsen skal enten leveres i Devilry, eller i obligkassa som står i gangen utenfor ekspedisjonen i 7. et. i Niels Henrik Abels hus. Frist er *kl. 14.30 torsdag 17/9*. Besvarelsen *skal* være skrevet av deg selv, for hånd eller på datamaskin.

Studenter som blir syke eller av andre grunner trenger å søke om utsettelse for denne obligatoriske oppgaven, må ta kontakt med studieadministrasjonen ved Matematisk institutt (7. et. Niels Henrik Abels hus, telefon 22 8558 88, e-post: studieinfo@math.uio.no) i god tid før innleveringsfristen.

Du kan lære masse fra medstudenter så det er både lurt og lærerikt å samarbeide med andre i arbeidet med oppgavene. Gruppelærerne har også anledning til å hjelpe, men ikke med ferdige løsninger. Målet med den obligatoriske oppgaven er at den skal bidra til læring, men *den endelige besvarelsen som du leverer må du skrive selv, og du må kunne redegjøre for innholdet ved en eventuell muntlig høring (aktuelt ved mistanke om ren avskrift, det gir dessverre ikke så mye læring :-))*.

Husk at de to obligatoriske oppgavene i MAT-INF 1100 begge må bestås for å kunne gå opp til endelig eksamen i kurset. *For å få bestått på denne første obligatoriske oppgaven må du gjøre seriøse løsningsforsøk på alle oppgavene, og minst to av de fire oppgavene bør være riktig besvart.*

Oppgaver

Oppgave 1. Uttrykk følgende tall som en sifferutvikling i 2-tallsystemet, det vil si skriv dem på formen $(d_n d_{n-1} \ldots d_1 d_0 \,.\, d_{-1} d_{-2} \ldots)_2$. Angi den repeterende sekvensen i de tilfellene der sifrene gjentar seg.

a) $\mathrm{da7}_{16}$

b) $29/64$

c) $1/25$

d) 0.4_{16}

e) $\mathrm{b2c}_{16}/1000_{16}$.

Oppgave 2. Følgen $\{x_n\}$ er gitt ved differensligningen

$$x_n = \big(\cos(x_{n-1})\big)^2 \sin(x_{n-2}) \quad \text{for } n \geq 2$$

og $x_0 = \pi/2$ og $x_1 = 3$. Vis ved induksjon at $0 \leq x_n \leq 1$ for alle heltall $n \geq 2$.

Oppgave 3. I denne oppgaven skal vi løse annengradsligninger på datamaskin.

a) Skriv et Python-program som regner ut de reelle røttene til annengradspolynomet $ax^2 + bx + c = 0$, og skriver de ut. Du kan bruke formlene

$$x_1 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}, \quad x_2 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

for de to løsningene. Koeffisientene a, b og c er reelle tall, og skal leses inn fra terminalen. Programmet ditt trenger ikke ta høyde for at a kan være 0, eller at du kan få et negativt tall under rottegnet. Test programmet ditt med tre eksempler som du også løser manuelt for å sjekke at det er riktig.

b) Bruk programmet i a) til å løse ligningen

$$10^{-8}x^2 + 10x + 10^{-8} = 0$$

(i terminalen kan du skrive `1e-8` i stedet for `10**(-8)`, for at Python skal klare å tolke tallene som flyttall). En av røttene x_1, x_2 som maskinen regner ut vil få stor relativ avrundingsfeil på datamaskinen. Hvilken av røttene gjelder dette, og hva skyldes det? Foreslå en alternativ formel for denne roten som unngår problemet.

Oppgave 4. Følgende python-program er gitt:

```
from random import random

antfeil = 0; N = 10000
x0 = y0 = z0 = 0.0
feil1 = feil2 = 0.0

for i in range(N):
    x = random(); y = random(); z = random();
    res1 = (x + y)**3
    res2 = x**3 + 3*x**2*y + 3*x*y**2 + y**3

    if res1 != res2:
        antfeil += 1
        x0 = x; y0 = y; z0 = z
        feil1 = res1
        feil2 = res2

print (100. * antfeil/N)
print x0, y0, z0, feil1 - feil2
```

En kjøring av programmet ga utskriften

```
50.64
0.82693218052 0.276620385833 0.244855430623 -4.4408920985e-16
```

a) Forklar hva programmet gjør og hva utskriften forteller oss.

b) Endre programmet slik at det i stedet sammenligner de to størrelsene `x+(y+z)` og `(x+y)+z`, og kjør det på nytt. Du skal nå se at det første tallet som blir skrevet ut er lavere enn det første tallet som ble skrevet ut over (d.v.s. `50.64`). Kan du tenke deg en mulig forklaring på dette?

Lykke til!!