

## Tentamen: Programmering med PYTHON, MVG301.

**Tid och plats:** Må 9/6 2022 kl 8:30–12.30

**Ansvarig lärare:** Katarina Blom. Tel 772 10 97

**Betygsgränser:** G: 12p, VG: 18p, maximalt 24p

**Hjälpmedel:** Inga hjälpmedel (ett formelblad finns bilagt tentan).

Lösningar kommer att läggas ut på kurshemsidan första arbetsdagen efter tentamenstillfället. Resultat meddelas via epost från LADOK.

---

- 1 Man kan bestämma en approximation till integralen  $\int_a^b f(x)dx$  genom att beräkna summan (4p)

$$h \cdot \sum_{i=0}^{n-1} f(x_i)$$

där  $h = \frac{b-a}{n}$  och  $x_i = a + i \cdot h$ . Skriv ett program i PYTHON som bestämmer en approximation till integralen

$$\int_{-1}^1 e^{-x^2} dx$$

genom att beräkna summan ovan. Låt  $n = 100$ .

- 2 Produkten  $\mathbf{a}_i \cdot \mathbf{b}_j$  mellan en kolumnvektor  $\mathbf{a}_i$  och en radvektor  $\mathbf{b}_j$  kallas yttre produkt. (4p)  
T.ex. produkten mellan

$$\mathbf{a}_i = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix} \text{ och } \mathbf{b}_j = \begin{bmatrix} 10 & 20 \end{bmatrix}$$

blir

$$\mathbf{a}_i \cdot \mathbf{b}_j = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix} \cdot \begin{bmatrix} 10 & 20 \end{bmatrix} = \begin{bmatrix} 1 \cdot 10 & 1 \cdot 20 \\ 3 \cdot 10 & 3 \cdot 20 \\ 5 \cdot 10 & 5 \cdot 20 \end{bmatrix} = \begin{bmatrix} 10 & 20 \\ 30 & 60 \\ 50 & 100 \end{bmatrix}$$

Skriv en funktion i PYTHON som beräknar yttre produkten mellan en radvektor och en kolumnvektor. Låt funktionen ha de två vektorerna som inparametrar. Du kan utgå från att den första inparametern är kolumnvektorn och den andra är radvektorn.

- 3 Matrisprodukten mellan  $m \times k$  matrisen  $\mathbf{A}$  och  $k \times n$ -matrisen  $\mathbf{B}$  kan beräknas genom att (4p)  
summera de  $k$  yttre produkterna mellan kolonnerna i  $\mathbf{A}$  och raderna i  $\mathbf{B}$ .

T.ex. produkten mellan

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 3 & 6 \end{bmatrix} \text{ och } \mathbf{B} = \begin{bmatrix} 10 & 20 \\ 30 & 40 \end{bmatrix}$$

kan beräknas genom att summera de yttre produkterna

$$\begin{aligned} \mathbf{AB} &= \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 3 & 6 \end{bmatrix} \begin{bmatrix} 10 & 20 \\ 30 & 40 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ 3 \end{bmatrix} \begin{bmatrix} 10 & 20 \end{bmatrix} + \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix} \begin{bmatrix} 30 & 40 \end{bmatrix} = \\ &= \begin{bmatrix} 10 & 20 \\ 30 & 60 \\ 30 & 60 \end{bmatrix} + \begin{bmatrix} 60 & 80 \\ 120 & 160 \\ 180 & 240 \end{bmatrix} = \begin{bmatrix} 70 & 100 \\ 150 & 220 \\ 210 & 300 \end{bmatrix} \end{aligned}$$

Skriv en funktion i PYTHON som beräknar produkten av två matriser genom att addera yttre produkter enligt ovan. Låt funktionen ha två inparametrar, matriserna som ska multipliceras. Om dimensionerna på matriserna inte stämmer, så att matrismultiplikation inte är definierad, ska funktionen returnera en tom matris (`[]`). Använd funktionen du skrivit i uppgift 2 i din lösning. (Om du inte löst uppgift 2, kan du anta att det finns en sådan funktion).

4 På bilden ser vi planet

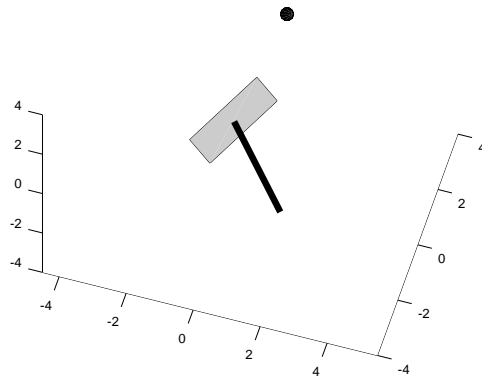
(4p)

$$2x - 2y - z = 0$$

på intervallet  $-1 \leq x \leq 1$  och  $-1 \leq y \leq 1$ , planets normal

$$\begin{cases} x(t) = 2t \\ y(t) = -2t \\ z(t) = -t \end{cases} \quad 0 \leq t \leq 1$$

och punkten  $(x, y, z) = (1, 2, 3)$ . Skriv ett program som ritar planet, normalen och punkten.



5 Skriv en funktion som returnerar hur många gånger orden `typ` och `liksom` förekommer i en textfil. Funktionen ska ha en inparameter, ett filnamn, och den ska returnera antal förekomster av orden. Du kan anta att filen innehåller ett ord per rad. (4p)

6 Låt  $n$  vara ett positivt heltal. Bestäm alla positiva heltal som delar  $n$  och bilda summan av dessa delare. I vissa fall kommer denna summa att vara ett primtal. Skriv ett program som skriver ut de femton minsta talen  $n$  med denna egenskap. (4p)

Exempel:  $n = 4$  har delarna 1, 2, 4 och summan av dessa tal är 7, som ju är ett primtal.  $n = 5$  har delarna 1, 5 och summan, 6, är inte ett primtal.

Låt datorn göra jobbet, du ska inte göra någon matematisk härledning. Använd gärna funktionen `isprime` från paketet `sympy` i din lösning. Anropet `isprime(n)` returnerar `True` om  $n$  är ett primtal. Om  $n$  inte är ett primtal returnerar funktionen `False`.