

DANMARKS TEKNISKE UNIVERSITET

KURSUSNAVN	INTRODUKTION TIL PROGRAMMERING OG DATABEHANDLING
KURSUSNUMMER	02631, 02692, 02694, 02633
HJÆLPEMIDLER	ALLE HJÆLPEMIDLER ER TILLADT
VARIGHED	2 TIMER
VÆGTNING	OPGAVERNE VÆGTES ENS

INDHOLD

ASSIGNMENT A: EXPONENTIAL SERIES EXPANSION	2
ASSIGNMENT B: ALPHA TO PHONE NUMBER	3
ASSIGNMENT C: GUESS THE GENDER	4
ASSIGNMENT D: BIRTHDAY PROBLEM	5
ASSIGNMENT E: MATRIX SEARCH	6

AFLEVERING

Du skal aflevere dine løsninger elektronisk:

1. Du kan uploade dine løsninger individuelt på CodeJudge (dtu.codejudge.net/prog-f16/assignment) under *Afleveringer/Exam*. Når du afleverer en løsning på CodeJudge bliver det test-eksempel som fremgår i opgavebeskrivelsen kørt på din løsning. Hvis din løsning består denne ene test, vil den fremgå som *Submitted*. Det betyder at din løsning består på dette ene test-eksempel. Du kan uploade til CodeJudge så mange gange som du ønsker under eksamen.
2. Du skal uploade alle dine løsninger på CampusNet. Hver assignment skal uploades som en separat .py fil med samme navn som funktionen i opgaven:
 - (a) `eseries.py`
 - (b) `alphaToPhone.py`
 - (c) `genderGuess.py`
 - (d) `birthday.py`
 - (e) `matrixSearch.py`

Filerne skal afleveres separat (*ikke* som en zip-fil) og skal have præcis disse filnavne.

Efter eksamen vil dine løsninger blive automatisk evalueret på CodeJudge på en række forskellige tests for at undersøge om de generelt fungerer korrekt. Bedømmelsen af din aflevering foretages udeklukkende på baggrund af hvor mange af de automatiserede test der består.

- Du skal sikre dig at din kode følger specifikationerne nøje.
- Hver løsning må ikke indeholde nogen yderligere kode udover den specificerede funktion.
- Husk at du kan tjekke om dine løsninger følger specifikationerne ved at uploade dem til CodeJudge.
- Bemærk at alle vektorer og matricer anvendt som input eller output skal være numpy arrays.

Assignment A Exponential series expansion

Exponentialfunktionen e^x kan approximeres med følgende potensrække:

$$f(x) = \sum_{i=0}^{N-1} \frac{x^i}{i!}$$

hvor $i!$ betegner fakultets-funktionen af i , og N er antallet af led.

■ **Problemdefinition**

Skriv en funktion med navnet `eseries` der tager som input x og N , og evaluerer udtrykket ovenfor.

■ **Løsningsskabelon**

```
def eseries(x, N):  
    #insert your code  
    return f
```

Input

x x -værdi hvor approximationen skal evalueres (decimal tal).
N Antallet af led (positiv heltal).

Output

f Approximationen af eksponentialfunktionen af x (decimaltal).

■ **Eksempel**

Forestil dig at evaluere $f(x)$ ved $x = 1.23$ med $N = 5$ led. Leddene kan beregnes som (her vist med fem decimaler)

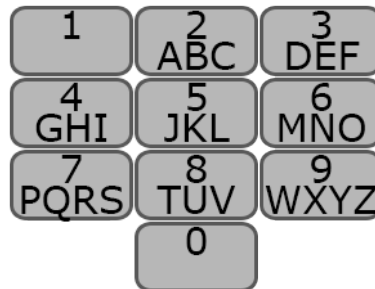
$i = 0$	$i = 1$	$i = 2$	$i = 3$	$i = 4$
1.00000	1.23000	0.75645	0.31014	0.09537

Summen kan da beregnes til 3.39196 hvilket er det endlige resultat.

A

Assignment B Alpha to phone number

På en telefons numeriske tastatur er hvert bogstav i alfabetet tildelt et af cifrene 2-9. Det gør det muligt at skrive alfa-numeriske telefonnumre som et mix af bogstaver og tal (ved at erstatte tal i telefonnummeret med tilsvarende bogstaver).



■ Problemdefinition

Skriv en funktion med navnet `alphaToPhone` der tager som input et alfa-numerisk (bogstaver og tal) telefonnummer som en streng, og returnerer det tilsvarende numeriske (kun tal) telefonnummer som en streng. Du kan antage at alle bogstaver i input er givet som store bogstaver.

■ Løsningsskabelon

```
def alphaToPhone(alpha):  
    #insert your code  
    return phone
```

Input

alpha Alfa-numerisk telefonnummer (streng indeholdende bogstaver og tal).

Output

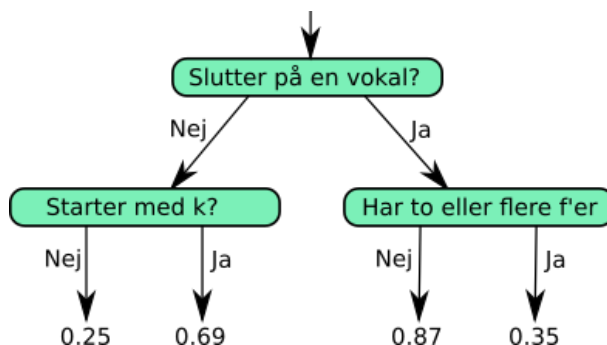
phone Numerisk telefonnummer (streng kun indeholdende tal).

■ Eksempel

Forestil dig følgende alfa-numeriske telefonnummer 4525DTU1. Omdannet til et numerisk telefonnummer bliver det til 45253881.

Assignment C Guess the gender

Du er givet et simpelt “beslutningstræ” nedenfor, som har til formål at forudsige en persons køn på baggrund af hans eller hendes navn.



De fire tal der er opgivet ved de mulige udfald er sandsynligheden for at det givne navn er et pigenavn.

■ Problemdefinition

Skriv en funktion med navnet `genderGuess` der tager som input et navn som en streng og beregner sandsynligheden for at navnet er et pigenavn baseret på beslutningstræet ovenfor. Du kan antage at input-navnet kun består af de små bogstaver `a–z`. Vi betegner bogstaverne `a`, `e`, `i`, `o`, `u`, og `y` som vokaler.

■ Løsningsskabelon

```
def genderGuess(name):  
    #insert your code  
    return pFemale
```

Input

`name` Navn (streng).

Output

`pFemale` Sandsynligheden for at navnet er et pigenavn (decimaltal).

■ Eksempel

Forestil dig navnet `affonso`. Vi starter i toppen af beslutningstræet. Da navnet slutter på en vokal, `o`, går vi ned til højre i beslutningstræet. Derefter, da navnet indeholder to eller flere `f`'er, går vi ned til højre igen. Det endelige resultat, som aflæses ved pilen, er at sandsynligheden for et pigenavn er `0.35`.

C

Assignment D Birthday problem

Det såkaldte “fødselsdagsproblem” består i at beregne sandsynligheden for at to (eller flere) personer i en population på n personer har samme fødselsdag. Denne sandsynlighed kan beregnes som:

$$P(n) = 1 - \exp(\ln\Gamma(k+1) - \ln\Gamma(k-n+1) - n \log(k))$$

hvor $\log(\cdot)$ og $\exp(\cdot)$ er den naturlige logaritme og eksponentialfunktion, og $\ln\Gamma(\cdot)$ er den såkaldte log-gamma funktion, som er implementeret i Python som funktionen `math.lgamma`. Tallet k er antallet af dage i et år, hvilket vi sætter til $k = 365$, og vi kan antage at $2 \leq n \leq k$.

■ Problemdefinition

Skriv en funktion med navnet `birthday` der tager som input størrelsen på populationen, n , og returnerer sandsynligheden $P(n)$ for at to (eller flere) personer i populationen har samme fødselsdag.

■ Løsningsskabelon

```
def birthday(n):  
    #insert your code  
    return P
```

Input

n Antallet af personer i populationen (positivt heltal).

Output

P Sandsynligheden for at to (eller flere) personer har samme fødselsdag (decimaltal).

■ Eksempel

Forestil dig en population med størrelsen $n = 23$. Sandsynligheden kan beregnes som følger (vist med fire decimaler):

$$P(23) = 1 - \exp(\ln\Gamma(365+1) - \ln\Gamma(365-23+1) - 23 \log(365)) \quad (1)$$

$$= 1 - \exp(1792.3316 - 1657.3419 - 135.6976) \quad (2)$$

$$= 0.5073 \quad (3)$$

Du er givet en matrix med heltal, hvor både rækkerne og søjlerne er sorteret i stigende orden. Matricen kunne for eksempel se ud som følger:

$$\begin{bmatrix} 1 & 2 & 6 & 10 \\ 3 & 7 & 7 & 13 \\ 7 & 9 & 11 & 14 \end{bmatrix}.$$

Lad os kalde matricen A , dens dimensioner $M \times N$, og dens elementer $a_{i,j}$. Den følgende algoritme er en effektiv måde at undersøge om og hvor et specifikt tal, x , findes i matricen:

1. Begynd i øverste højre hjørne af matricen, $i = 1, j = N$.
2. Se på tallet $a_{i,j}$:
 - (a) Hvis $a_{i,j} = x$, er du færdig. Returner resultatet $[i, j]$.
 - (b) Ellers, hvis $a_{i,j} > x$ gå et skridt til venstre, $j \leftarrow j - 1$.
 - (c) Ellers, hvis $a_{i,j} < x$ gå et skridt ned, $i \leftarrow i + 1$.
3. Hvis du er inden for matricen, dvs. $i \leq M$ og $j > 0$, gentag fra 2. Ellers, returner resultatet $[0, 0]$.

■ Problemdefinition

Skriv en funktion med navnet `matrixSearch` der tager som input en matrix A som beskrevet ovenfor samt et tal x der skal ledes efter i matricen. Funktionen skal returnere en vektor med koordinaterne $[i, j]$ for det første sted x findes i matricen ifølge ovenstående algoritme. Hvis x ikke findes i matricen, skal funktionen returnere vektoren $[0, 0]$.

■ Løsningsskabelon

```
def matrixSearch(A, x):
    #insert your code
    return index
```

Input

A	Række- og søjle-sorteret matrix ($M \times N$) med heltal.
x	Tal der skal ledes efter (heltal).

Output

index	Række- og søjle-koordinater for det fundne tal (vektor med længde 2). Returner $[0, 0]$ hvis tallet ikke findes i matricen.
--------------	--

■ Eksempel

Forestil dig at vi leder efter talet $x = 7$ i matricen ovenfor hvor $M = 3$ og $N = 4$. Vi begynder i øverste højre hjørne ved $a_{1,4} = 10$. Da $10 > x$ går vi til venstre til $a_{1,3} = 6$. Da $6 < x$ går vi ned til $a_{2,3} = 7$. Da dette tal er lig x , returnerer vi resultatet $[2, 3]$.