

# DMA

*Ugeopgave 1*

BEATE BERENDT SØEGAARD  
MATHIAS LARSEN  
SIMON ROTENDAHL

Datalogi

8. september 2016

```

from __future__ import division
import math

A = [1,5,6,10,15,16,17,23]
B = [5,10,17,6,23,15,16,1]

def exists(A,n,x):
    """
    Checks wheter or not x exists
    in the list before hi >= lo,
    if yes then it'll return true
    otherwise return false.
    """
    lo = 0
    hi = n
    while hi >= lo:
        mid = int(math.floor((hi+lo)/2))
        if x > A[mid]:
            lo = mid+1
        elif x < A[mid]:
            hi = mid-1
        else:
            return True
    return False

print "exists(A,8,17) returnerer: ",exists(A,8,17)
print "*" * 20
print "exists(A,8,14) returnerer: ",exists(A,8,14)
print "*" * 20
print "exists(A,4,16) returnerer: ",exists(A,4,16)
print "*" * 20
print "exists(A,8,3) returnerer: ",exists(A,8,3) # mid 4, mid 1,
mid 0
print "*" * 20
print "exists(B) returnerer: ",exists(B,6,12)
print "*" * 20
print "exists(B,7,1) returnerer: ",exists(B,7,1)

```

## Del 1

**a**

Den returnerer True, da dels er  $x = 17$  og algorithmen når at køre igennem før hi er lig med lo.

**b**

Den returnerer False, da  $x = 14$  ikke er en del af listen.

**c**

Den returnerer False, da hi ikke er stor nok til at algorithm kan nå, at køre igennem listen.

**d**

Vi for mid til at være hhv.:

- 4
- 1
- 0

## Del 2

Tjekker hvorvidt  $x$  eksistere i listen, hvis ja så returnerer den True. Den returnerer False dels hvis  $x$  ikke eksisterer i listen, men også når hi er for lavt da den ikke når at køre igennem listen. Listen skal også være sorteret ellers giver den et tilfældigt svar.

## Del 3

**a**

Den kan ikke give True på noget som helst tidspunkt, da  $x$  skal eksistere i listen da mid er et indekserings tal.

**b**

Ja, den giver False når hi er for lavt og når et lavt tal har et højere indeks, og vice versa, end mid så vil algorithmen returnerer False.

## Del 4

Når  $n = 17$  kan vores implementeret algoritme, skrevet i python, ikke køre da vi vil få en *out of bounce* fejl. I andre tilfælde, såsom C vil du bare få skrald, hvilket kan være hvad som helst.