Norges teknisk–naturvitenskapelige
universitet
Institutt for datateknikk og
informasjonsvitenskap

1 Massemidtpunkt

Kodesnutt 1

```python
def center_of_mass(stang):
    total_weight = sum(stang)
    halfway = total_weight / 2
    weight_so_far = 0

    i = 0
    while (weight_so_far + stang[i]) <= halfway:
        weight_so_far = weight_so_far + stang[i]
        i = i + 1

    center = i
    center = center + (halfway - weight_so_far) / stang[i]
    return center


print(center_of_mass([1])) # 0.5
print(center_of_mass([1, 1])) # 1
print(center_of_mass([1, 1, 1])) # 1.5
print(center_of_mass([3, 1, 3])) # 1.5
print(center_of_mass([1, 2, 3, 4])) # 2.6667
```

2 Løkker

Kodesnutt 2

a)
```python
import math

def is_prime(number):
    for divider in range(2, number - 1):
        if number % divider == 0:
            return False
    return True
```

Kodesnutt 3

b)
```python
def separate(numbers, threshold):
    list1 = []
    list2 = []
    for i in numbers:
        if i < threshold:
            list1.append(i)
        else:
            list2.append(i)
    return list1, list2
```

Kodesnutt 4

c)
```python
def multiplication_table(n):
    multiplication_table = []
    for y in range(1, n + 1):
        table = []
        for x in range(1, n + 1):
            table.append(x * y)
        multiplication_table.append(table)
    return multiplication_table

print(multiplication_table(5))
```

3 Strenghåndtering

Kodesnutt 5

```python
def compare(string1, string2):
    length = len(string1)

    if length == len(string2):
        for i in range(length):
            if string1[i] != string2[i]:
                return False
        return True
    return False

def reverse(string):
    reversed_string = ''

    for i in range(len(string)-1, -1, -1):
        reversed_string += string[i]

    return reversed_string

def palindrome(string):
    return string == reverse(string)

def contains(string1, string2):
    for x in range(len(string1)):
        if string1[x:(len(string2)+x)] == string2:
            return True
    return False
```