

```

1  ## 1.1
2  def digit_sum(n):
3      s_digits = list(str(n))
4      dsum = 0
5      for s_digit in s_digits:
6          dsum += int(s_digit)
7      return dsum # deze regel ontbrak
8
9  def is_harshad(n):
10     return n % digit_sum(n) == 0
11
12  ## 1.2
13  # O(log(n)) want loop over digits van n, wat schaalt met de logaritme.
14  # Een 10x grotere n geeft 1 extra iteratie.
15
16
17  ## 1.3
18  d = 8
19  e = 2
20  from math import * # Hier wordt alles geïmporteerd uit de math module, dus ook
21                      # pi en e. Vanaf nu heeft e dus de waarde 2.71828... .
22  sqrt(d ** e)
23
24
25  ## 1.4
26  a = 1 < 2 or 4 > 2          # True want (1 < 2) or (4 > 2)
27  b = not 1 < 2 or 4 > 2     # True want (not (1 < 2)) or (4 > 2)
28  c = not (1 < 2 or 4 > 2)   # False
29  d = not 0 < 1              # False want not (0 < 1)
30  e = 4 > 2 or 10/0 == 0     # True want (4 > 2) is True waardoor OR altijd True zou
31  opleveren en alles daarna niet meer geevalueerd wordt
32
33  ## 1.5
34  def pascal(n):
35      if n == 1:
36          return [1]
37      else:
38          previous_line = pascal(n-1)
39          line = [1]
40          for i in range(len(previous_line)-1):
41              line += [previous_line[i] + previous_line[i+1]]
42          line += [1]
43      return line
44
45  print(pascal(5))
46
47
48  # Dit is een recursieve functie. Ik noem previous_line even pl in deze uitleg
49  # pascal(1)=[1]
50  # pascal(2)=[1, 1]          (pl=[1], dus geeft [1] + [1], want for loop
51  wordt 0 x uitgevoerd)
52  # pascal(3)=[1, 2, 1]      (pl=[1,1], dus geeft [1]+ [pl[0] + pl[1] ] +
53  [1] = [1] + [1+1] + [1] = [1,2,1])
54  # pascal(4)=[1, 3, 3, 1]  (pl=[1,2,1], dus geeft [1] + [pl[0] + pl[1]]
55  + [pl[1] + pl[2]] + [1] = [1] + [1+2] + [2+1] + [1])
56  # pascal(5)=[1, 4, 6, 4, 1] (pl=[1,3,3,1], dus geeft [1] + [pl[0] +
57  pl[1]] + [pl[1] + pl[2]] + [pl[2] + pl[3]] + [1] = [1] + [1+3] + [3+3] + [3+1] + [1])
58
59  #Antwoord: [1,4,6,4,1]
60
61
62
63
64
65
66

```

```

67  ## 2a
68  def volgend_hagelsteengetal(n):
69      if n%2==0:
70          return n//2 # NB n/2 geeft een float terug - dit reken ik niet fout.
71      return 3*n+1
72
73  ## 2b
74  def hagelsteenrij(n):
75      try:
76          if n!=int(n) or n<1:
77              return []
78      except ValueError: # Andere methoden om foutieve input te checken (type(n) is
79                          # int) zijn uiteraard ook goed. Het programma crashen op onjuiste aanroep niet.
80          return []
81
82      rij=[n]
83      while n>1:
84          n=volgend_hagelsteengetal(n)
85          rij += [n]
86
87      return rij
88
89  ## 2c
90  def rijeigenschappen(n):
91      for i in range(1, n+1):
92          rij = hagelsteenrij(i)
93          print(i, len(rij), max(rij))
94
95  ## 2c zonder max functie te gebruiken
96  def rijeigenschappen(n):
97      for i in range(1, n+1):
98          rij = hagelsteenrij(i)
99          mx = rij[0]
100         for getal in rij:
101             if getal > mx:
102                 mx = getal
103         print(i, len(rij), mx)
104
105  ## 2d
106  def voldoet_aan_Collatz(n):
107      rij=[]
108      while n>1:
109          if n in rij:
110              return False
111          rij += [n]
112          n=volgend_hagelsteengetal(n)
113      return True
114
115  ## 2d zonder "if ... in" constructie:
116  def voldoet_aan_Collatz(n):
117      rij=[]
118      while n>1:
119          for getal in rij:
120              if n == getal: #herhaling
121                  return False
122          rij += [n]
123          n=volgend_hagelsteengetal(n)
124      return True
125
126  ## 2d met dictionary
127  def voldoet_aan_Collatz(n):
128      used_nums={n: None}
129      while n>1:
130          n=volgend_hagelsteengetal(n)
131          if n in used_nums:
132              return False
133          used_nums[n]=None;
134      return True
135
136

```

```
137  ## 2e
138  def vind_tegenvoorbeeld(n):
139      while check_Collatz(n):
140          if n%1000 == 0:
141              print(n)
142              n += 1
143      print ("Tatadataa! De reeks die start met",n, "voldoet niet aan Collatz! Het is
144          dus toch niet waar!")
145
146
```