Let op: De nadruk van deze toets ligt op de verdiepende hoofdstukken 11 t/m 13. Zie Python toets I voor vragen gericht op hoofdstukken 1 t/m 7, en Python toets II voor hoofdstukken 5 t/m 10.

|  |
| --- |
| **Toets Informatica**  **Programmeren met Python III**    Stof:   * Basiskennis van Python * (Eigen) Functies & parameters * Tekstbestanden * Strings * Geneste loops en geneste lijsten   Schrijf je antwoorden op een **apart papier**. Een gewone (niet-grafische) rekenmachine is toegestaan.  Deze toets bestaat uit **X opgaven**. **Norm**: Bij elke vraag staat aangegeven hoeveel punten je maximaal voor die vraag kunt behalen. **Totaal XX punten.**  **Succes!** |

[2pt] Opgave 1:

Geef voor ieder van de volgende aan of het WAAR of ONWAAR is:  
a) Als je een bestand open maakt om te lezen, en het bestand bestaat nog niet, dan zal een nieuw bestand aangemaakt worden.

b) Als je een bestand open maakt om in te schrijven, en het bestand bestaat nog niet, dan zal een nieuw bestand aangemaakt worden.

c) Als je een bestand open maakt om in te schrijven, en het bestand bestaat al, dan zal over het oorspronkelijke bestand overschreven worden.

[2pt] ANTWOORD Opgave 1

1. FALSE: Het programma geeft een fout
2. TRUE
3. TRUE

[-1pt per fout]

[1pt] Opgave 2:

Om een bestand als geheel in te lezen als één string, dan gebruiken we:

1. bestand.read()
2. bestand.readline()
3. bestand.readlines()
4. open(bestandsnaam, “r”)

[1pt] ANTWOORD OPGAVE 2

a

[2pt] Opgave 3:

De volgende vraag gaat over de code hieronder. Je mag er vanuit gaan dat getallen een lijst met gehele getallen is.

|  |
| --- |
| def functie1(getallen):  antwoord = [ ]  for getal in getallen:  if getal >= 0:  antwoord = antwoord + [getal]  return antwoord |

Vat kort samen wat het doel van ‘functie1’ is. Let op: Er wordt gevraagd om een samenvatting, dus niet een uitleg van elke regel.

[2pt] ANTWOORD Opgave 3:

levert een lijst op met alleen de getallen groter of gelijk aan 0

Omdat het qua syntax veel op som van getallen lijkt (als je [ ] over het hoofd ziet) wordt maar minimale aftrek hiervoor gegeven.: -1pt

Als ‘positieve getallen’ wordt geinterpreteerd als inclusief 0 (terwijl dat eigenlijk niet zo is): geen punten aftrek

Bij het expliciet vermelden ‘getallen groter dan 0’: -1 punt

[4pt] Opgave 4:

De volgende vraag gaat over de code hieronder. Je mag er vanuit gaan dat getallen een lijstje getallen is.

|  |
| --- |
| def functie2():  a = getallen[0]  for waarde in getallen:  if waarde > a:  a = waarde  return a  def functie3(a):  b = 0  for waarde in getallen:  if waarde == a:  b += 1  return b  c = functie2()  d = functie3(c)  print(d) |

a) Wat is de uitvoer van ‘functie2’ als getallen = [2,-4,7,4,7]

b) Vat kort samen wat het doel van het hele programma is. Let op: Er wordt gevraagd om een samenvatting, dus niet een uitleg van elke regel.

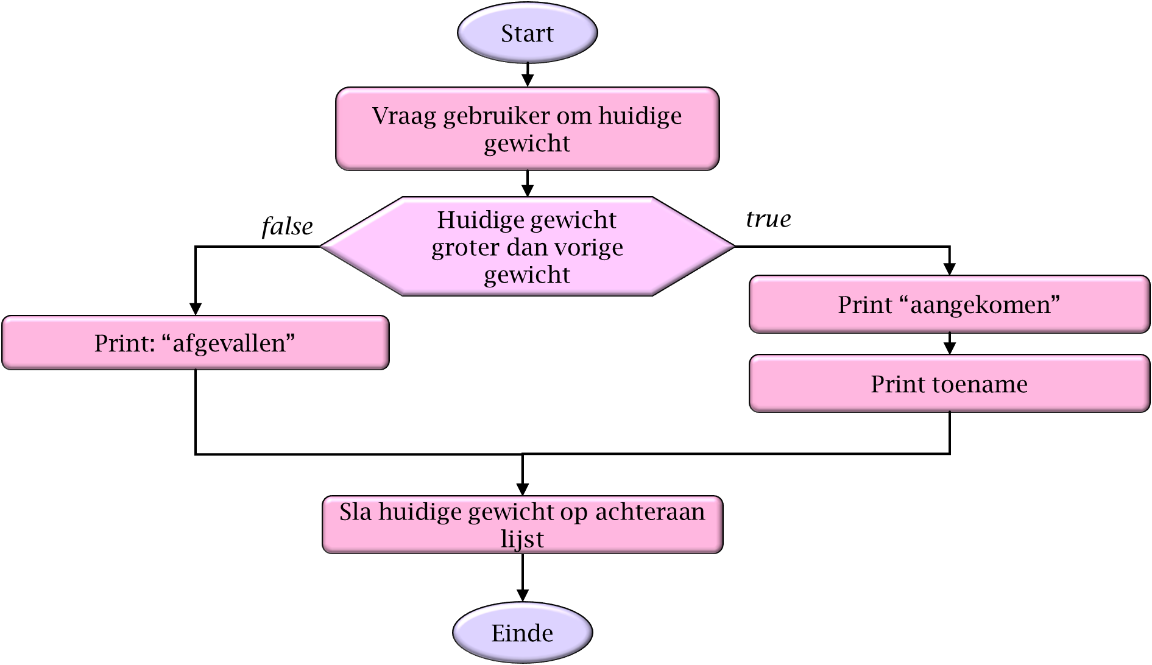
Tips: Ga na wat het doel van ‘functie2’ is. Ga na wat het doel van ‘functie3’ is. Wat is hun doel samen als geheel?

[4pt] ANTWOORD Opgave 4:

1. [1pt] 7 (want fun1 bepaalt het maximum)
2. [3pt] Het drukt af hoe vaak [1pt] de grootste getal [1pt] voorkomt in een niet-lege lijst[1pt].

[4pt] Opgave 5:

Een atleet schrijft een programma om zijn gewicht (in kilo’s) bij te houden. Voor het onthouden wordt een lijst gebruikt, telkens wordt de gewicht achteraan de lijst toegevoegd. Hieronder zie je een stroomdiagram voor dat programma. Vertaal het stroomdiagram naar Python code. Je kunt uitgaan van deze lijst met de gewichten van drie metingen: gewichtenLijst = [56.3, 54.3, 58.9]. De laatste gemeten gewicht kun je dus uit het lijst opvragen.



[4pt] ANTWOORD Opgave 5:

gewichtenLijst = [56.3, 54.3, 58.9]  
  
nieuwGewicht = float (input (**"Hoeveel weeg je?"**) )  
  
**if** nieuwGewicht > gewichtenLijst[ len(gewichtenLijst)-1 ]:  
 print(**"Je bent aangekomen"**)  
 aangekomen = nieuwGewicht - gewichtenLijst[ len(gewichtenLijst)-1 ]  
 print(aangekomen)  
  
**else**:  
 print(**"Je bent afgevallen"**)

gewichtenLijst += [nieuwGewicht]  
*# OF  
#gewichtenLijst.append(nieuwGewicht)  
  
#om te testen:  
#print(gewichtenLijst)*

-1 voor elk fout: nieuwgewicht, float, if, len, len-1, berekening, juiste prints, opslaan in lijst

[3pt] Opgave 6: Schrijf een programma dat berekent hoeveel het verven van een muur kost. Doe dat in de volgende drie stappen:

* Schrijf een functie berekenOpp dat twee getallen als **parameter** meekrijgt, namelijk de lengte en de hoogte van de muur, de oppervlakte berekent en deze met een **return** oplevert.
* Roep je functie aan met de getallen 3 en 5.
* Verf kost 4 Euro per vierkante meter. Om de kosten te berekenen gebruik je de code uit **deel b**: vermenigvuldig het antwoord van berekenOpp met 4. Druk de kosten vervolgens af.

[3pt] ANTWOORD Opgave 6: Let op: vraag komt ook voor in Python II toets

**def** calcArea(length, width):  
 area = length \* width  
 **return** area  
  
calculatedArea = calcArea(3,5)  
kosten = calculatedArea \* 4  
print( kosten )

a) [2pt] voor de definitie met parameters en return

b) [1pt] voor aanroep van calcArea(3,5) met juiste waarden en opslaan in een variabele: calculatedArea = calcArea(3,5)  
c) [1pt] berekening van kosten en opslaan in een variabele:

kosten = calculatedArea \* 4

[3pt] Opgave 7:

De volgende vraag gaat over de code hieronder.

|  |
| --- |
| def functie4(val):    for i in range(val):      for j in range(i):        print("\*", end="")      print() |

Vat kort samen wat het doel van ‘functie4’ is. Let op: Er wordt gevraagd om een samenvatting, dus niet een uitleg van elke regel.

[3pt] ANTWOORD Opgave 7:

Print een driehoek vak sterretjes: 4 rijen, met eerste rij 0 sterren, dan 1 … 3

Bij vierkant: - 1pt

Per verkeerde/ontbrekende stop criteria: -1pt

Voor het omschrijven als rij/vierkant/…maar verkeerde stopcriteria: 1pt toekennen

[1pt] Opgave 8:

Wat tekent de onderstaande code? De turtle begint altijd gericht naar rechts.

|  |  |
| --- | --- |
| **import** turtle  turtle.pendown() **for** x **in** range(4):  **for** y **in** range (360):  turtle.forward(1)  turtle.right(1)  turtle.penup()  turtle.forward(100)  turtle.pendown() | 1. 4 cirkels naast elkaar 2. 4 cirkels onder elkaar 3. 360 vierkanten 4. 4 vierkanten onder elkaar |

[1pt] Antwoord Opgave 8:

a. 4 cirkels naast elkaar

[5pt] Opgave 9:

Schrijf een programma dat, gegeven een lijst met getallen, een staafdiagram met sterretjes afdrukt. Bijvoorbeeld bij [1,3,0,2,4] wordt afgedrukt:

\*

\*\*\*

\*\*

\*\*\*\*

[5pt] ANTWOORD OPGAVE 9

lijst = [1,3,0,2,4]

for getal in lijst:

  for i in range(getal):

    print("\*", end="")

  print()

for getal in lijst:

  aantal\_sterren = 0

  while aantal\_sterren < getal:

    print("\*", end="")

    aantal\_sterren+=1

  print()

of

for x in lijst:

print(x\*"\*", end="")

print()

OF

for y in (lijst):

  aantal\_ster = y \* '\*'

  print(aantal\_ster)

[6pt] Opgave 9:

|  |
| --- |
| woord = (“fiets\nbel”)  print(woord) |

1. Wat wordt er afgedrukt?
2. Wat levert woord.find(“ts”) op?
3. Hoe druk je de eerste letter van woord af?
4. Zonder gebruik te maken van een cijfer, hoe druk je de laatste letter van woord af? Je mag in je antwoord de getallen van 0 t/m 10 dus niet gebruiken.
5. Schrijf een functie dat woord als parameter binnenkrijgt en het woord in hoofdletters oplevert.

[6pt] ANTWOORD Opgave 9

1. [1pt] Over twee regels afgedrukt:

fiets

bel

1. [1pt] 3
2. [1pt] woord[0]
3. [2pt] woord[len(woord)-1]
4. [2pt]

def maakHoofdletters(woord):

hoofdletter\_woord = woord.upper()

return hoofdletter\_woord