ICTS op 7,5 van de 20 punten:

MK1: 2X3 pixels en 18 bits, hoeveel kleuren kan je hebben?

A. geen van onderstaande
B. 2
C. 8 (x2)(18/(2x3) = 3 => 2^3 = 8)
D. 16 (x1)

MK2: verband HTTP en HTML
= HTML is tekst en HTTP zorgt ervoor dat het op internet geraakt

MK3: wat is fout

A. actieve RFID tags kunnen op korte afstand werken (8x)
B. RFID tags kunnen door een smartphone met NFC gelezen worden
C. RFID tags zijn vergelijkbaar met het scannen van barcodes
D. Een stelling over een koffer met een RFID tag, maar die was zeker juist

MK4: wat is juist

A. thin client bevat business logic (deze is het niet)
B. iets dat clients rechtstreeks kunnen communiceren met database services
C. op fat client is software gemakkelijk te beheren

MK5: Welke stelling is fout?

A. SCM zorgt voor een verhoging van de voorraad (2x)

MK6: Drie belangrijkste dingen die een informatiesysteem doet met data

a) Verwerking, opslag en communicatie (5x)
b) Verwerking, opslag en software
c) invoer, uitvoer, verwerking (1x)
d) SCM, CRM

MK7: vraag zoals deze maar andere cijfers:
Hoeveel bytes heb ik nodig voor een YouTube video van 60 seconden met een resolutie van 4K, 30 fps als framerate en er 24 bits gebruikt worden voor de weergave van de kleur ?

A) Dit is niet te berekenen

B) 4K*30*60*24 bytes

C) 4K*30*60*24*8 bytes

D) 4K*30*60*3 bytes

ANTWOORD D

MK8: Correcte query iets met winkel, prijs, product

vreemde sleutel winkel verwijsde naar (winkel,product,prijs)

A. SELECT … FROM… WHERE… AND … AND … AND … (2X)
B. SELECT…FROM winkel WHERE…. (1X°

MK9: bedrijf X en Y zijn een XML structuur overeengekomen (X plaatst bestelling bij Y en Y moet controleren of het de correcte structuur heeft ofzoiets)

A. X en Y hebben een akkoord over de structuur van de XML schema (2x)
B. andere optie: controleren met een basis schema
C. ook optie met sleutels denk ik?

PROGRAMMEREN:

(MK op 2,5p van de 8p)

vraag 1: Welke uitspraak is juist?

- De tijd-complexiteit van een algoritme geeft aan hoe het geheugengebruik schaalt in functie van de grootte van de invoer.
- De ruimte-complexiteit van een algoritme geeft aan hoe de uitvoeringstijd schaalt in functie van de grootte van de invoer.
- De snelheid van een algoritme hangt af van de tijd-complexiteit van een algoritme
- De snelheid van een algoritme hangt af van de kloksnelheid van de processor en de tijd-complexiteit van een algoritme (x4)

vraag 2: iets over programma's en processen, foute stelling kiezen

- een foute algoritme zou eventueel tot een correcte uitkomst kunnen leiden
- een correct algoritme dat buiten de voorwaarde van het eindig proces werkt is desalniettemin een correct algoritme (2x)

vraag 3:wat is fout

- else clausule is altijd exclusief (7x)
- else clausule is altijd exhaustief
- for loop kan met voorspellende iteratie
- while loop kan met voorspellende iteratie

vraag 4: wat is fout (programma over het controleren van kolommen van een matrix was gegeven)

- lange tekst over dat je correctheid kan controleren met lege matrix, eenheidsmatrix... en dat dit de correctheid niet exact kan aangeven (10x)
- break zorgt ervoor dat het stopt met de elementen te vergelijken en vervolgens verder lijn 12-15 uitvoert en dus print dat de matrix niet symmetrisch is (1x)
- Er is mogelijks een efficiëntere methode om de matrix te overlopen/iets over dat het sommige elementen op 2 keer doet(op positie [i][j] en [j][i]) (5x) , dit is toch fout want for lus gaat 1 keer over elk element lopen van de matrix?

ANTWOORD A

uitleg: volgens mij is het eerste juist want die heeft dat in de les gezegd volgens mij (of op het forum), er was een soortgelijke vraag op een herhalingstaak => stress testing

Er was toch wel een efficiëntere code (bv break in de buitenste lus? zie pwp slide 23 van hdfst 7) of was da ni hetzelfde op het examen? denk wel da da hetzelfde als op het examen was.

Ik denk dat het programma a[0][0] = a[0][0] zou controleren, dat geeft (1=1) => ok, verder controleren
a[0][1] = a[1][0], dus 2=2 => ok
a[0][2] =? a[2][0], 3 is niet gelijk aan 5, dus stoppen met vergelijken en direct naar de conclusie gaan. Dus er zijn geen elementen die 2 keer(op positie [i][j] en [j][i]) gecontroleerd werden en kan deze stelling niet juist zijn? Er zijn miss efficiëntere methodes, maar dat deel van 2 keer controleren is sws fout. Dus de laatste is fout?

Wouter Verbeke

RE: hoofdstuk 7 toets oef 1

Beste Malin,

Met stress testing kan je inderdaad controleren of een algoritme correct werkt, maar ben je niet helemaal zeker of het algoritme werkelijk in alle mogelijke gevallen correct functioneert - omdat je nu eenmaal niet voor alle mogelijke gevallen (dat kunnen er oneindig zijn) kan controleren of de output correct is. Vandaar dat in antwoord B staat 'mogelijks de enige manier'.

Antwoord C: fouten in een programma kan je soms op dergelijke manier opsporen, maar van zodra een programma groter wordt, dan is het niet vanzelfsprekend om alle regels code na te lezen en te controleren (denk aan een programma met honderden functies). Als mensen maken we snel foutjes hierbij, dus dit is zeker niet altijd een handige en ook geen sluitende manier om fouten op te sporen.

Mvg,

Wouter

23 mei 2023 8:42:57 CEST
Totaalaantal weergaven: 56 (Uw weergaven: 3)

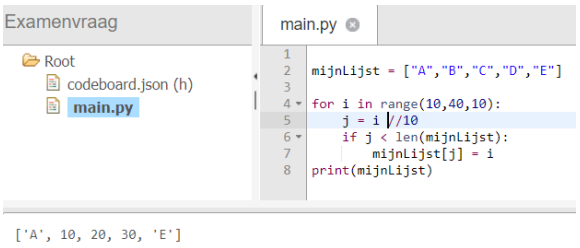
Dus de eerste stelling is fout? → ja ik denk het en dus het juiste antwoord

vraag 5: wat is fout over modules

- modules maken programmeren minder complex (1x)
- module zorgt ervoor dat je in een functie minder argumenten moet aanroepen (7x)

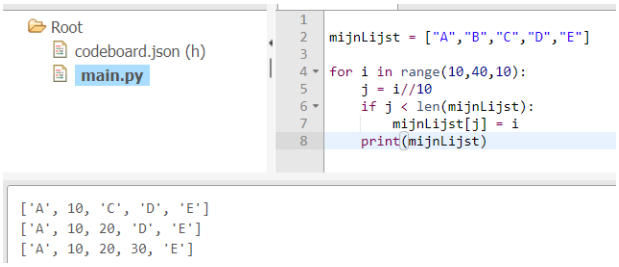
(open vragen op 52 punten herleid naar 6p van de 8,5p)

open vraag: (3punten)
geef de output van volgende code:
Als ik die code ingeef kom ik op de code hieronder uit? (in het zwarte kader)




```
['A', 10, 20, 30, 'E']
```

Was het niet dit???? ik ook de print stond toch uit de if-lus?



openvraag:
iets me numpy aanvullen (eenheidsmatrix creëren adhv bijlage => vul ... aan)
import numpy as np
from random import randrange
print(np.identity(randrange(6,10))
.
nog 2 lijnen code waar al het begin gegeven was

Openvraag:
Creëer een functie die een matrix maakt met de dimensie: een aantal rijen en kolommen gegeven door de gebruiker, opgevuld door paramVulling gekozen door de gebruiker.

- functie "maken":
  - input: paramRij, paramKolom, paramVulling
- functie"weergeven": dimensie paramRij x paramKolom opgevuld met paramVulling

```
def afmetingVragen(paramTekst):
        print("aantal %s" %paramTekst)
        afstandStr = input("Geef een afstand in)
        while not afstandStr.isnumeric():
                print("incorrecte waarde")
                afstandStr = input("Geef een afstand in")
        afstand = int(afstandStr)
        return afstand
```

(dit is wat ik had ongeveer)
moest je nie nog een functie maken: afmetingenVragen()? Komt op etzelfde neer denkik

```
def maken( paramVulling, paramRij, paramKolom):
        matrix = [ ]
        for i in range(len(paramRij)):
                rij = [ ]
                for j in range(len(paramKolom)):
                        rij += [paramVulling]
                matrix += [rij]
        return matrix

def weergeven(paramMatrix):
.       for rij in paramMatrix:
                print(rij)
```

dan nog de main() functie samenstellen maar vgm was er al een deel gegeven

open vraag 2:

lijst met klanten gegeven

gegevens =[[ID, inkomen, #jaar klant, spaargeld], [....],[....],....]

namen = {"naam" : ID, ...}

1) de rang bepalen voor elke klant + toevoegen aan de lijst 'gegevens'
2) woordenboek maken die de kosten bepaalt aan de hand van de rang van de klant
   kosten = {"N":2 , "P":1, "S":0.5}
3) met Boolean onderzoeken of een persoon klant is van de bank (functie schrijven)
4) naam van een persoon gegeven en dan kijken of die klant is of niet, als die klant is moet je de kosten bepalen aan de hand van voorgaande stukken code + output printen (X is een N klant en betaald 2% kosten ofzoiets) (.format(paramNaam, rang, kosten))
5)
6)