

Bi2a-O Kans 1 Correctiemodel

Begindatum: donderdag 25 januari 2018, 9.00 uur

Uiterlijke inleverdatum: donderdag 1 februari 2018, 9.00 uur

Mijn correctiemodel gaat uit van het principe dat je voor ieder onderdeel de volle punten krijgt, en je punten aftrek krijgt op het moment dat er iets aan ontbreekt of niet klopt.

Bonuspunten tellen altijd alleen maar voor het aantal punten van dat onderdeel, dus ze stapelen niet over het gehele tentamen!

Beoordeling

De beoordeling van het totale tentamen is als volgt.

Criterium			Max punten	Beoordeling
1	Python Code (30)			
	a.	Commentaar	10	
	b.	Onderverdeling in functies	10	
	c.	Datatypes en variabelen	10	
2	Juiste werking (70)			
	a.	Parsen bestand naar dictionary	10	
	b.	Werken met dictionaries	10	
	c.	Regular expressions	10	
	d.	Matplotlib	10	
	e.	Exception handling	20	
	f.	Pickling	10	
Totaal			100	

1.a. Commentaar (10pt)

Documenteer je code goed, door aan het begin van je programma een omschrijving te geven met auteur, datum, versie en functie programma, en door in je code een korte omschrijving te geven van wat iedere functie doet.

- -10: Commentaar afwezig
- -7; commentaar sumier, kort en/of weinig verhelderend
- -3; commentaar an sich voldoende aanwezig, maar ontbrekende stukjes, zoals auteur en functie input/output

1.b. Onderverdeling in functies (10pt)

Je deelt je code goed op in functies met parameters en return statements.

- -10; geen functies of alleen een main()
- -5; functies zijn verantwoordelijk voor teveel dingen (denk aan single responsibility)

Op dit onderdeel kan je extra punten verdienen.

- +5; gebruik van object orientatie (op de juiste manier)

1.c. Datatypes en variabelen (10pt)

Dit onderdeel gaat over het gebruiken van de juiste datatypes en namen voor je variabelen, ofwel hoe netjes je omgaat met variabelen.

- -10; naamgeving door middel van enkele letters (a = [], b = 0.5, etc)
- -5; naamgeving met vage namen (lijst1 = [], var1 = 0.5, etc)
- -3; gebruikt verkeerde datatypes voor wat de opdracht is

2.a. Parsen bestand naar dictionary (10pt)

Het bestand wordt ingelezen en omgezet naar een bruikbare dictionary met behulp van een of meerdere functies. Het is voor dit onderdeel niet belangrijk of je de inhoud van de dictionary ook er uit kan krijgen, daar toetst het volgende onderdeel op. Belangrijk is dat de inhoud van de dictionary overeenstemt met het bestand.

- -10; leest bestand niet
- -5; leest bestand op meerdere plaatsen in de code in terwijl dit redundant is
- -5; leest bestand, maar leest niet correct
- -5; leest bestand, maar zet dit niet om naar een dictionary

Het komt vrijwel nooit voor, maar in theorie zou je 0 punten op dit onderdeel kunnen verdienen met bovengenoemde puntentelling. Op dit onderdeel kunnen ook extra punten worden verdient.

- +5; bestand wordt ingelezen via een GUI.

2.b. Werken met dictionaries (10pt)

Dit onderdeel beoordeelt hoe goed je de inhoud van een dictionary verder kan verwerken nadat deze gevuld is. Ook belangrijk voor dit onderdeel is dat je verderop nog gebruik maakt van je gemaakte dictionary in het vorige onderdeel.

- -10; geen gebruik dictionary, dictionary wordt aangemaakt, maar daarna nooit meer geraadpleegd.
- -7; onlogische key::value structuur
- -5; onjuiste manier van ophalen van informatie uit dictionary

2.c. Regular expressions (10pt)

Regular expressions worden beoordeeld in dit onderdeel.

- -10; geen regular expressions
- -5; onjuiste regular expression op basis van de zoektermen
- -5; onjuist aanroepen/inzetten regular expression functionaliteit

Op dit onderdeel kunnen ook extra punten worden verdient.

- +5; je maakt gebruik van recursieve functies.

2.d. Matplotlib (10pt)

Voor het maken van een grafiek maak je gebruik van Matplotlib. Zoals eerder genoemd is het niet belangrijk dat de grafiek perse een bepaalde vorm heeft (bijv een pie chart), maar wel belangrijk voor dit onderdeel is dat de grafiek de juiste informatie toont aan de hand van de opdracht.

- -10; geen grafiek
- -5; onjuiste data in grafiek, ofwel, grafiek toont niet het gevraagde

Op dit onderdeel kunnen ook extra punten worden verdient.

- +5; je maakt gebruik van een GUI en toont de grafiek hierin.

2.e. Exception handling (20pt)

Exception handling is een belangrijk onderdeel van programmeren. Vandaar dat dit onderdeel ook extra zwaar meetelt t.a.v. de andere onderdelen. Denk goed na over welke errors je wilt afvangen op welke plek. Het is voor dit onderdeel niet verplicht dat je ook errors opgooit!

- -20; geen exception handling
- -15; exception handling met alleen een algemene except, niet nagedacht over wat er precies afgevangen moet worden
- -10; exception handling alleen in de main()
- -5; maakt gebruik van het afvangen van meerdere exceptions, maar heeft nog niet goed duidelijk welke exceptions voor dat onderdeel niet van toepassing zijn (bijv. IOError/FileNotFoundError afvangen terwijl er nergens een bestand gelezen wordt binnen die functie)

Op dit onderdeel kunnen ook extra punten worden verdient.

- +5; errors worden afgevangen en met de gebruiker gedeelt via pop-ups o.i.d. in de GUI.
- + 5; gooit zelf errors op waar toepasbaar.

2.f. Pickling (10pt)

Je datastructuur wordt gepickled en ontpickled. Bij het opstarten van het programma controleert de code of er al een pickle is en zoniet, dan maakt het alsnog een nieuwe dictionary en pickled deze.

- -10; geen gebruik van pickle
- -5; schrijft weg maar leest niet in (of andersom voor zover mogelijk)
- -5; leest niet correct in.

Ook hier zijn extra punten te verdienen.

- +5; pickling gebeurt met behulp van object orientatie