

NAAM:

volgnummer:

Informatica Labotest 2 mei 2022 - Python

Richtlijnen

Je mag enkel methodes en functies gebruiken die vermeld zijn op de API. Wat daarbuiten valt (zelfs synoniemen) wordt als foutief beschouwd.

Gebruik zinvolle namen voor variabelen en eventuele hulpfuncties.

Type-hinting moet gebruikt worden bij definities van functies.

Imports hoeft je nergens te schrijven. Er wordt nergens foutafhandeling gevraagd.

Er zijn twee vragen, die onafhankelijk zijn van elkaar.

Veel Succes!!

Vraag 1

Gegeven het hoofdprogramma (links), en twee mogelijke outputs daarvan (rechts).

```
oorspronkelijk = "Dit is een zin!"
print(f"oorspronkelijk: {oorspronkelijk}")
onleesbaar = onleesbare_versie(oorspronkelijk, "Dei")
print(f"onleesbaar: {onleesbaar}")
opnieuw leesbaar = leesbare_versie(onleesbaar)
print(f"opnieuw leesbaar: {opnieuw leesbaar}")
```

```
oorspronkelijk: Dit is een zin!
onleesbaar:      68s105st 105us 101@101~n z105fn!
opnieuw leesbaar: Dit is een zin!
```

```
oorspronkelijk: Dit is een zin!
onleesbaar:      68z105ht 105gs 101<101?n z105dn!
opnieuw leesbaar: Dit is een zin!
```

Als je de oorspronkelijke en de onleesbare tekst met elkaar vergelijkt, kom je tot deze conclusie:

- Elk karakterteken uit "Dei" werd in de oorspronkelijke tekst vervangen door zijn ascii-waarde.
- Na die ascii-waarde komt nog een willekeurig karakterteken. Dat zal ervoor zorgen dat er nooit twee ascii-waardes vlak na elkaar voorkomen (anders zou het niet duidelijk zijn waar het ene ascii-getal ophoudt en het andere start).

De omgekeerde bewerking maakt de tekst weer leesbaar: alle getallen worden vervangen door het bijhorende karakterteken - en uiteraard wordt het willekeurige teken dat een getal afsluit, genegeerd.

In deze opdracht wordt de implementatie gevraagd van de twee functies uit het hoofdprogramma, maar eerst komt er een hulpfunctie.

Let op! Uiteraard moeten deze functies zo geschreven worden, dat ook een hoofdprogramma met andere parameters volgens bovenstaande principes werkt.

1. Definieer een functie `random_letter()` die een willekeurig karakterteken teruggeeft. Dit karakterteken is een leesteken (!, ?, @, enz.) zoals gedefinieerd in de gekende string-constante van Python, of een kleine of grote letter uit het alfabet.

OPLOSSING:

2. Definieer een functie `onleesbare_versie(tekst, opsomming)` die een gegeven tekst omvormt tot een onleesbare tekst. Alle karakters uit de tekst die in de tweede parameter te vinden zijn (de opsomming), worden vervangen door hun ascii-waarde gevolgd door een willekeurig karakterteken (zoals gedefinieerd in de voorgaande functie). De onleesbaar gemaakte tekst wordt teruggegeven.

OPLOSSING:

KLADRUIMTE: (TREK ZELF EEN KADER ALS HIER EEN OPLOSSING KOMT)

3. Definieer een functie `leesbare_versie(tekst)` die de omgekeerde bewerking doet. De gegeven tekst bevat lettertekens en cijfers. Vervang de groepjes van cijfers (die een getal vormen) door het karakterteken dat dit getal als ascii-waarde heeft. Let op: na elk getal komt nog een willekeurig karakterteken, dat je mag negeren. De leesbaar gemaakte tekst wordt teruggegeven.

OPLOSSING: