

Oefeningen opgaves les 3

Oefening 1

Schrijf een programma dat 2 functies bevat: `berekenEindTijd(begin, duur)` en `berekenVerschilTijd(begin, einde)`.

De variabelen `begin`, `duur` en `einde` zijn strings die een tijd bevatten in het formaat `UUxMM` of `UxMM`, met 'U' het uur, 'M' de minuten en 'x' een willekeurig scheidingsteken. Dus `02h45`, `3:18`, `15u22` zijn allemaal geldige vormen.

Als `begin` en `duur` gegeven zijn, gebruik je `berekenEindTijd(begin, duur)` om de eindtijd weer te geven, bijvoorbeeld `begin = 10,00` en `duur = 1,15`, dan krijg je `11:15` terug.

Als `begin` en eindtijd gegeven zijn, dan geeft `berekenVerschilTijd(begin, einde)` de tijd tussen de beide terug. Dus `10,00` en `11,15` geeft `01:15` terug. Opgelet, als de eindtijd vroeger is dan de begintijd, dan gaan we er van uit dat we middernacht passeren. Dus met `begin = 10u00` en `eind = 2u00` krijgen we `04:00` terug.

Oefening 2

Maak opnieuw een programma om gemiddeldes te berekenen, zoals in één van de vorige oefeningen.

Schrijf een functie die alle getallen die je ingeeft controleert.

De getallen die mogelijk moeten zijn, kunnen positief of negatief zijn, van het type integer en float. Je moet ook getallen met wetenschappelijke notatie kunnen verwerken (zoals 3,14e3 bijvoorbeeld). Echter, de normale wetenschappelijke notatie laat geen float toe voor de exponent, dus 3,14e2,5 is niet toegelaten. Ons programma accepteert dergelijke invoer wel, aangezien de e gewoon staat voor 10^{exponent} en $10^{2,5}$ kan wel degelijk (probeer maar).

Oefening 3

Maak een programma met 2 functies, de ene zet een getal om naar romeinse cijfers en de andere zet romeinse cijfers om naar een gewoon getal.

Gebruik alleen positieve gehele getallen en het maximum is 3999, omdat je anders met een bovenlijn moet gaan werken. Zie tabel hieronder.

Wanneer je functies geschreven zijn, schrijf je 3 functionaliteiten:

- je schrijft code die het langst mogelijke romeinse getal zoekt (dit is het getal met het meeste symbolen).
- Je schrijft code die je toelaat om een getal in te geven en die je dan het romeinse equivalent teruggeeft. Maak daar een lus van, zodat je meer dan 1 getal kunt ingeven. Aanvaard alleen integers.
- Hetzelfde als het vorige, alleen geef je nu romeinse cijfers in en krijg je een getal terug.

Laat de gebruiker kiezen welke functionaliteit hij gebruikt. Steek die functionaliteit eventueel ook in functies.

Romeinse cijfersymbolen:

I	V	X	L	C	D	M
1	5	10	50	100	500	1000

Normaal		Met bovenlijn	
IV	4	$\overline{\text{IV}}$	4,000
V	5	$\overline{\text{V}}$	5,000
VI	6	$\overline{\text{VI}}$	6,000
VII	7	$\overline{\text{VII}}$	7,000
VIII	8	$\overline{\text{VIII}}$	8,000
IX	9	$\overline{\text{IX}}$	9,000
X	10	$\overline{\text{X}}$	10,000
L	50	$\overline{\text{L}}$	50,000
C	100	$\overline{\text{C}}$	100,000
D	500	$\overline{\text{D}}$	500,000
M	1000	$\overline{\text{M}}$	1,000,000

Oefening 4

Herschrijf het kassa programma voor de chocolade winkel uit de vorige serie oefeningen, maar nu met gebruikmaking van functies. Verbeter de gebruiksvriendelijkheid door een code te vragen voor het soort chocolade, en dan pas het aantal. Je hoeft niet voor alle 3 soorten repen een getal in te geven als de klant maar 1 of 2 soorten koopt. Idem voor leveranciers.

Zet eventueel je code online op github en zet een link in de commentaar op de laatste video van les 3.