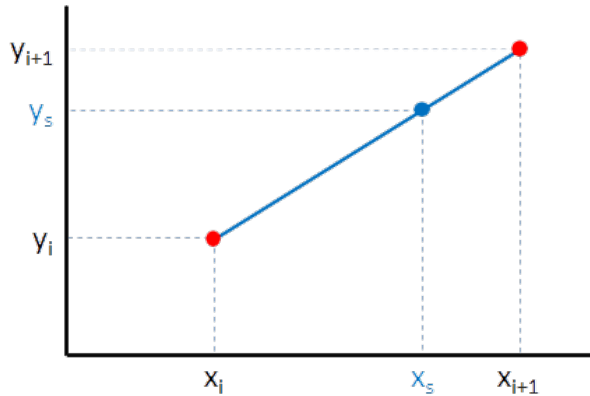


Linear interpolation

We have a table containing the y -values of a certain measurement, which match the x -values 1, 2, 3, ..., 100. To estimate the y_s result that matches the x -value x_s — which lies between the successive whole x -values x_i and x_{i+1} — we use the formula for *linear interpolation*: $y_s = y_i + (x_s - x_i) \frac{(y_{i+1} - y_i)}{(x_{i+1} - x_i)}$



example of linear interpolation

Assignment

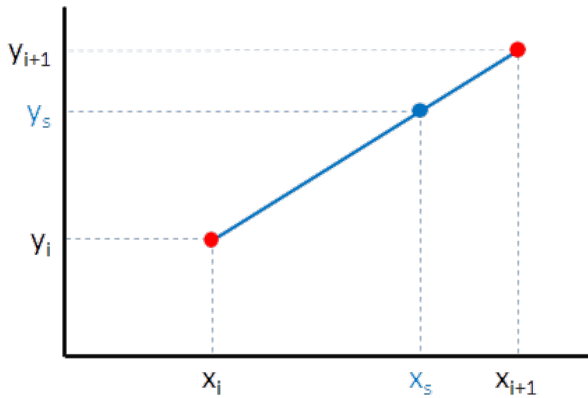
1. Write a function `linearInterpolation` that takes two arguments. The first argument must be a list with 100 integers and the second argument must be a float between 1 and 100. The list represents the y -values which match the x -values 1, 2, ..., 100. The function returns the y -value which corresponds with the x -value given by the second argument. If the arguments do not meet the right conditions, an error message must be printed and the value `None` must be returned. Look at the example below to understand these error messages.

Example

```
>>> linearInterpolation(list(range(1, 101)), 10.75)
10.75
>>> linearInterpolation(list(range(1, 101))[:-1], 10.75)
90.25
>>> linearInterpolation([42] * 100, 10.75)
42.0
>>> linearInterpolation(42, 10.75)
Expected type: <class 'list'>, Received type: <class 'int'>
>>> linearInterpolation(42, 0)
Expected type: <class 'list'>, Received type: <class 'int'>
>>> linearInterpolation([42], 10)
Expected type: <class 'float'>, Received type: <class 'int'>
>>> linearInterpolation([42], 10.75)
Expected length: 100, Received length: 1
>>> linearInterpolation([42] * 100, 100.75)
Interpolation is only possible for the values 1 up to and including 100.
```

We beschikken over een tabel, waarvan de gegevens de y -waarden voor een bepaalde meting voorstellen. Daarin staan de meetresultaten die horen bij de x -waarden 1, 2, 3, ..., 100. Om te schatten wat het resultaat y_s is dat hoort bij een x -waarde x_s — die tussen twee

opeenvolgende gehele x -waarden x_i en x_{i+1} in ligt — maken we gebruik van de formule voor *lineaire interpolatie*: $y_s = y_i + (x_s - x_i) \frac{(y_{i+1} - y_i)}{(x_{i+1} - x_i)}$



voorbeeld van lineaire interpolatie

opgave

1. Schrijf een functie `lineaireInterpolatie` die twee argumenten neemt. Het eerste argument moet een lijst met 100 getallen zijn en het tweede argument moet een float zijn tussen 1 en 100. De lijst stelt de y -waarden voor die horen bij de x -waarden $1, 2, \dots, 100$. De functie geeft de y -waarde terug die correspondeert met de x -waarde die gegeven wordt door het tweede argument. Indien de argumenten niet aan de juiste voorwaarden voldoen, dan wordt een foutboodschap afgedrukt en de waarde `None` wordt teruggegeven. Bekijk het onderstaande voorbeeld om deze foutboodschappen te achterhalen.

Voorbeeld

```
>>> lineaireInterpolatie(list(range(1, 101)), 10.75)
10.75
>>> lineaireInterpolatie(list(range(1, 101))[:-1], 10.75)
90.25
>>> lineaireInterpolatie([42] * 100, 10.75)
42.0
>>> lineaireInterpolatie(42, 10.75)
Verwachte type: <class 'list'>, Ontvangen type: <class 'int'>
>>> lineaireInterpolatie(42, 0)
Verwachte type: <class 'list'>, Ontvangen type: <class 'int'>
>>> lineaireInterpolatie([42], 10)
Verwachte type: <class 'float'>, Ontvangen type: <class 'int'>
>>> lineaireInterpolatie([42], 10.75)
Verwachte lengte: 100, Ontvangen lengte: 1
>>> lineaireInterpolatie([42] * 100, 100.75)
Interpolatie is enkel mogelijk voor waarden van 1 tot en met 100.
```