

Vrae / Questions

1. Vir die algemene kwadratiese vergelyking

For the general quadratic equation

[60%]

$$ax^2 + bx + c = 0, \quad (1)$$

word die oplossings van x gegee deur

the solutions of x are given by

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}. \quad (2)$$

Skryf 'n Python-funksie `qsolve` wat drie numeriese parameters a , b en c (in dié volgorde) neem, en dan die oplossings van x in Vgl. (1) volgens Vgl. (2) terugstuur. U mag slegs ingeboude operatore en funksies gebruik. Die volgorde waarin u die oplossings terugstuur, maak nie saak nie. *Let wel:* Aanvaar dat u funksie getalle as parameters sal ontvang. Aanvaar ook dat daar nie enige komplekse oplossings sal wees nie, dit wil sê, die diskriminant in Vgl. (2) sal nie-negatief wees. Stoor u funksie in 'n lêer genaamd `quadratic.py`, en gebruik die program `test_quadratic.py`, beskikbaar in 'n argief op die kursuswebwerf, om u werk te toets.

Write a Python function `qsolve` that takes three numeric parameters a , b , and c (in this order), and then *returns* the solutions of x in Eq. (1) according to Eq. (2). You may only use built-in operators and functions. The order in which you return the solutions does not matter. *Note:* Assume that your function will receive numbers as parameters. Also assume that there will not be any complex solutions, that is, the discriminant in Eq. (2) will be nonnegative. Save your function to a file called `quadratic.py`, and use the program `test_quadratic.py`, available in an archive on the course website, to test your work.

2. Skryf 'n Python-funksie `sort3` wat drie heelgetalparameters neem, hulle in stygende volgorde sorteer, en hulle dan in dié gesorteerde volgorde terugstuur. *Let wel:* Om dinge regverdig te hou teenoor diegene sonder 'n rekenaarwetenskapagtergrond, mag u slegs konsepte wat ons reeds tydens die lesings behandel het in u oplossing gebruik—dit beteken spesifiek u mag nie vertakking (`if`), herhaling (`for` en `while`) of ingeboude sorteerfunksies of -metodes gebruik nie. Stoor u funksie in 'n lêer genaamd `sort3.py` en gebruik die program `test_sort3.py`, beskikbaar in 'n argief op die kursuswebwerf, om u werk te toets. WENK: Dink wiskundig, en gebruik die ingeboude `min`- en `max`-funksies.

Write a Python function `sort3` that takes three integer parameters, sorts them in ascending order, and then returns them in this sorted order. *Note:* To keep things fair for those without a background in computer science, you may only use concepts we have already covered during the lectures in your solution—specifically, it means you may not use branching (`if`), repetition (`for` and `while`), or built-in sorting functions or methods. Save your function to a file called `sort3.py` and use the program `test_sort3.py`, available in an archive on the course website, to test your work. *HINT:* Think mathematically, and use the built-in functions `min` and `max`.

[40%]