Programmieren 1 (PRG1) Übung 2

2.1

Auswertungsreihenfolge von oben (höchste Priorität) nach unten (niedrigste Priorität)

1. **(**4 >> 2**)**
2. 2**\*\***2
3. 15 **\*** 2
4. (4 >> 2) **/** 2\*\*2
5. 19 **%** 20
6. 15 \* 2 – (4 >> 2)
7. 4 **>>** 2 (Priorität irrelevant, da Klammern um 4 >> 2)
8. 2 \*\* 2 **<** 19 % 20

(Quelle: Programmierhandzettel 1)

2.2

Die Typen von Literalen können in Python mithilfe der type() Funktion überprüft werden.

True = bool (Boolean)

„Hallo Welt“ = str (String)

‚23.5‘ = str (String)

17+4j = complex (Komplexe Zahl)

19. = float (Float; Kommazahlen)

655321 = int (Integer)

2.3

i = 1 Ergebnis: 1.0

i = 5 Ergebnis: 2.236067977499978

i = 9 Ergebnis: 3.000000001396984

i = 12 Ergebnis: 3.464101615137755

i = 16 Ergebnis: 4.000000000000051

Die gegebene Funktion berechnet die Quadratwurzel des eingegebenen i. Hierzu verwendet Programm eine iterative Vorgehensweise, die auch als Heron-Verfahren oder babylonisches Wurzelziehen bezeichnet wird. Die relevante Gleichung in diesem Zusammenhang lautet:

xn+1 = ½ ( xn + (a/xn))

oder im Falle des angegebenen Programms:

a = (a + (i / b)) / 2

Das Programm iteriert die untersuchte Zahl i solange, bis (a – b) <= 0.001 ist („while“ Schleife). Da sich a und b im Laufe des Verfahrens von beiden Seiten (höher und niedrigere Werte) an die Quadratwurzel annähern, wird ihre Differenz genutzt, um das Programm bei einer Genauigkeit bis zur dritten Nachkommastelle abzubrechen.

(Quelle: <https://de.wikipedia.org/wiki/Heron-Verfahren>; Datum: 02.11.17; Uhrzeit: 18.30 Uhr)