

CSI 31 HW5 Solutions

Page 76-77 True/False questions

1. True
2. False
3. False, they are built-in operations
4. True
5. False, it computes the square root of a number
6. False, they have finite number of decimal places
7. True
8. True
9. True
10. False, $4+5$ is of type *int*, whereas $4.0+5.0$ is of type *float*

page 737 / Multiple Choice questions

1. **c)** rational
2. **d)** `sqrt()`
3. **d)** an import statement
4. **b)** $4! = 1*2*3*4 = 24$
5. **b)** float
6. **c)** 32, because _____ , total is $2^5=32$
7. **a)** floats to ints
8. **d)** `abs` (although it is tempting to choose `b)` `round` as the answer, the problem with `round` is when we call `round(1.45)` - it rounds it to the whole number and the type of return value is integer then)
9. **a)** accumulator, comment: `b)` counted loop - will not be correct as the answer, because this is a Python's structure that is used to accumulate factorial
10. **d)**

pages 74-75 / Discussion

1.

- a)** float, 7.4 **b)** float, 5.0 **c)** int, 8 **d)** exception: the radicand should be ≥ 0
e) int, 11

- 2. a)** $(3+4)*5$ **b)** $(n*(n-1))/2$ **c)** $4*\text{math.pi}*r**2$
d) $\text{math.sqrt}(r*\text{math.cos}(a)**2 + r*\text{math.sin}(b)**2)$ **e)** $(y2-y1)/(x2-x1)$

3.

- a)** `range(5)` will produce the sequence [0,1,2,3,4]
b) `range(3,10)` will produce the sequence [3,4,5,6,7,8,9]
c) `range(4,13,3)` will produce the sequence [4,7,10]
d) `range(15,5,-2)` will produce the sequence [15,13,11,9,7]
e) `range(5,3)` will produce empty sequence []

4

a)

1
4
9
16
25
36
49
64
81
100

b)

1 : 1
3 : 27
5 : 125
7 : 343
9 : 729

(c) show the output that would be generated by each of the following program fragments:

```
x=2
y = 10
for j in range(0,y,x):
    print(j,end="")
    print(x+y)
print("done")
```

x is assigned the value of 2,
y is assigned the value of 10,
then we have a counted for loop, with counted **j** going over the sequence **[0,2,4,6,8]**
1st iteration: j = 0, and **012** will be displayed
2nd iteration: j = 2, and **212** will be displayed
3rd iteration: j = 4, and **412** will be displayed
4th iteration: j = 6, and **612** will be displayed
5th iteration: j = 8, and **812** will be displayed
then we'll exit from the loop, and
done will be displayed

Hence, on the screen we will see:

```
012
212
412
612
812
done
```

d) ans = 0

for i in range(1,11) will generate the sequence [1,2,...,10]

```
1st iteration (i=1): ans = ans + i*i = 0 + 1*1 = 1    1
2nd iteration (i=2): ans = 1 + 2*2 = 5              2
3rd iteration (i=3): ans = 5 + 3*3 = 14             3
4th iteration (i=4): ans = 14 + 4*4 = 30            4
5th iteration (i=5): ans = 30 + 5*5 = 55           5
6th iteration (i=6): ans = 55 + 6*6 = 91           6
7th iteration (i=7): ans = 91 + 7*7 = 140          7
8th iteration (i=8): ans = 140 + 8*8 = 204         8
9th iteration (i=9): ans = 204 + 9*9 = 285         9
10th iteration (i=10): ans = 285 + 10*10 = 385    10
At the end the value of ans is displayed          385
```

output

