COSI 10A Recitation #3

Agenda: Return & Math Library

Review of “return” Keyword

- It "returns" the result (value of the expression following the return keyword) to the caller.

```
def addNumbers(a,b):
    return a + b

print(addNumbers(4, 5))
```

Output is 9

- A return statement ends the execution of a function call.

```
def addNumbers(a,b):
    return a + b
    print("this code is not reachable!")

print(addNumbers(4, 5))
```

- Statements after the return statement are not executed.

Output is still 9!

Review of Python Math Library

Overview

- Provides mathematical functions and constants to perform calculations.

Built-in Module:

- It is a standard library module, meaning it comes pre-installed with Python and requires no additional installation.

Usage:

- Import the library using “import math” before using its functions.

Performance:

- Functions in the math library are typically faster and more efficient than equivalent operations implemented in Python itself.

Function name	Description	Example	Result
`abs(`*value*`)`	absolute value	`abs(-308)`	`308`
`min(`*value1, value2*`)`	smallest of values	`min(7,2,4,3)`	`2`
`max(`*value1, value2*`)`	largest of values	`max(7,2,4,3)`	`7`
`round(`*value*`)`	nearest whole number	`round(3.647)` `round(3.647, 1)`	`4` `3.6`

Function name	Description	Example	Result
`math.ceil(`*value*`)`	rounds up	`math.ceil(2.13)`	`3.0`
`math.floor(`*value*`)`	rounds down	`math.floor(2.13)`	`2.0`
`math.log(`*value, base*`)`	logarithm	`math.log(2,2)`	`1.0`
`math.sqrt(`*value*`)`	square root	`math.sqrt(2)`	`1.414234…`
`math.pow(`*x, y*`)`	x to the power of y	`math.sqrt(5,2)`	`25.0`

Constant	Description
`math.e`	2.7182818...
`math.pi`	3.1415926...

Problem 1

In physics, the displacement of a moving body represents its change in position over time while accelerating.

- Given initial velocity v_0 in m/s, acceleration a in m/s^2, and elapsed time t in s, the displacement of the body is:
- Displacement = $v_0 t + \frac{1}{2} a t^2$

Write a method `displacement` that accepts v_0, a, and t and computes and returns the change in position.

- example: `displacement(3.0, 4.0, 5.0)` returns `65.0`

https://codeshare.io/NKNKrr

In physics, the displacement of a moving body represents its change in position over time while accelerating.

- Given initial velocity v_0 in m/s, acceleration a in m/s^2, and elapsed time t in s, the displacement of the body is:
- Displacement = $v_0 t + \frac{1}{2} a t^2$

Write a method `displacement` that accepts v_0, a, and t and computes and returns the change in position.

- example: `displacement(3.0, 4.0, 5.0)` returns `65.0`

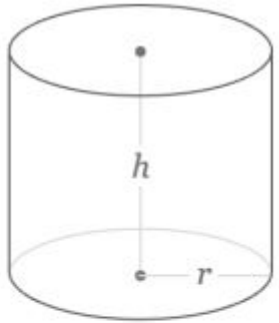
```
def displacement(v0, a, t):
    return v0 * t + 0.5 * a * t**2

print(displacement(3.0, 4.0, 5.0))
```

```
def displacement(v, a, t):
    d = v * t + 0.5 * a * t**2
    return d
```

Problem 2

The surface area of a cylinder

- surface area = $2\pi rh + 2\pi r^2$



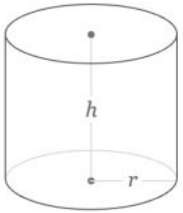
Write a method `calAreaCylinder` that accepts r, and h and computes and returns the surface area including the fractional part to two decimal points.

- example: `calAreaCylinder(4,10)` returns `351.86`

https://codeshare.io/NKNKrr

The surface area of a cylinder

- surface area = $2\pi rh + 2\pi r^2$



Write a method `calAreaCylinder` that accepts r, and h and computes and returns the surface area including the fractional part to two decimal points.

- example: `calAreaCylinder(4,10)` returns `351.86`

```python
def calAreaCylinder(r, h):
    surface_area = 2 * math.pi * r * h + 2 * math.pi * r**2
    return round(surface_area, 2)
```

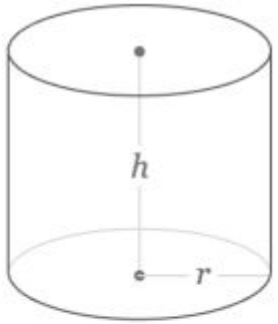
```python
def calAreaCylinder(r, h):
    return round(2 * math.pi * r * h + 2 * math.pi * r**2, 2)
```

```python
def calAreaCylinder(r, h):
    surface_area = round(2 * math.pi * r * h + 2 * math.pi * r**2, 2)
    return surface_area
```

Problem 3

The volume of a cylinder

- volume = $\pi r^2 h$



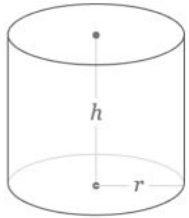
Write a method `calVolCylinder` that accepts r, and h and computes and returns the volume including the fractional part to two decimal points.

- example: `calVolCylinder(4,10)` returns `502.65`

https://codeshare.io/NKNKrr

The volume of a cylinder

- volume = $\pi r^2 h$



Write a method `calVolCylinder` that accepts r, and h and computes and returns the volume including the fractional part to two decimal points.

- example: `calVolCylinder(4,10)` returns `502.64`

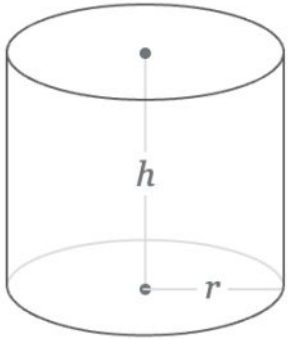
```
def calVolCylinder(r, h):
    v = math.pi * r**2 * h
    return round(v, 2)
```

```
def calVolCylinder(r, h):
    return round(math.pi * r**2 * h, 2)
```

```
def calVolCylinder(r, h):
    v = round(math.pi * r**2 * h, 2)
    return v
```

Problem 4

Write a program that calculates the volume of a cylinder.



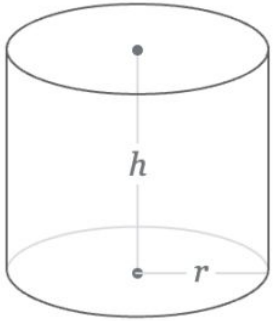
Your program should create two functions: 1) a function named calcArea(r) that accepts an integer number for the radius of a circle, calculate the area of the cross section using this formula $A=\pi r^2$, and returns the calculated area and 2) a function named calcVol(area, h) that accepts two parameters for the area of cross section and an integer number for the height, calculates the volume, and returns the calculated volume including the fractional part to two decimal points. Note that your programs need to get the input from users using `input` function.

When you run your program, the output should look exactly like the following:

```
Type integer number for radius:3
Type integer number for height:10
The volume for a cylinder with radius 3 and height 10 is
282.74
```

https://codeshare.io/NKNKrr

Write a program that calculates the volume of a cylinder.



Your program should create two functions: 1) a function named calcArea(r) that accepts an integer number for the radius of a circle, calculate the area of the cross section using this formula $A=\pi r^2$, and returns the calculated area and 2) a function named calcVol(area, h) that accepts two parameters for the area of cross section and an integer number for the height, calculates the volume, and returns the calculated volume including the fractional part to two decimal points. Note that your programs need to get the input from users using `input` function.

When you run your program, the output should look exactly like the following:

```
Type integer number for radius:3
Type integer number for height:10
The volume for a cylinder with radius 3 and height 10 is
282.74
```

```python
import math

def calcArea(r):
    return math.pi * r**2

def calcVol(area, h):
    return area * h

r = int(input("Type integer number for radius:"))
h = int(input("Type integer number for height:"))
area = calcArea(r)
volume = round(calcVol(area, h), 2)

print(f"The volume for a cylinder with radius {r} and height {h} is {volume}")
```

```
(base) jameskong@JimmyK-PC:~/cosi10aTA/Recitation 3$ /home/jameskc
Type integer number for radius:3
Type integer number for height:10
The volume for a cylinder with radius 3 and height 10 is 282.74
```

Problem 5

Write a program which calculates the monthly payment `p` for a loan using the following formula:

$$p = \frac{rP}{n\left[1 - \left(1 + \frac{r}{n}\right)^{-nt}\right]}$$

$$I = npt - P$$

Your program should have a function named `calcPayment(P, r, t, n)` that accepts four parameters and return the calculated monthly payment including the fractional part to two decimal points.
It should ask the user for:
the principal `P` (note this is a capital `P`, don't get it confused with the payment amount `p`)
the annual interest rate `r` (float number in percentage)
the number of years `t`
the number of payments per year `n` (typically 12)

Note that your programs need to get the input from users using `input` function and all input numbers should be integers except the annual interest rate.

When you run the script, the output should look exactly like the following:

```
Amortization Calculator
Principal, P = 400000
Annual Interest rate (10%), r = 6.5
Number of years, t = 30
Number of payments per year, n = 12
The monthly payment will be p = 2528.27
```

https://codeshare.io/NKNKrr

Hint: notice the user inputted annual interest rate is in percentage form...

Write a program which calculates the monthly payment `p` for a loan using the following formula:

$$p = \frac{rP}{n\left[1 - \left(1 + \frac{r}{n}\right)^{-nt}\right]}$$

$$I = npt - P$$

Your program should have a function named `calcPayment(P, r, t, n)` that accepts four parameters and return the calculated monthly payment including the fractional part to two decimal points.
It should ask the user for:
the principal `P` (note this is a capital `P`, don't get it confused with the payment amount `p`)
the annual interest rate `r` (float number in percentage)
the number of years `t`
the number of payments per year `n` (typically 12)

Note that your programs need to get the input from users using `input` function and all input numbers should be integers except the annual interest rate.

When you run the script, the output should look exactly like the following:

```
Amortization Calculator
Principal, P = 400000
Annual Interest rate (10%), r = 6.5
Number of years, t = 30
Number of payments per year, n = 12
The monthly payment will be p = 2528.27
```

```python
def calcPayment(P, r, t, n):
    monthlyRate = r / 100
    return (monthlyRate * P) / (n * (1 - (1 + monthlyRate / n)**(-n * t)))

print("Amortization Calculator")

principal = int(input("Principal, P = "))
rate = float(input("Annual Interest rate (10%), r = "))
years = int(input("Number of years, t = "))
payments = int(input("Number of payments per year, n = "))

monthlyPay = round(calcPayment(principal, rate, years, payments), 2)

print(f"The monthly payment will be p = {monthlyPay}")
```

```
(base) jameskong@JimmyK-PC:~/cosi10aTA/Recitation 3$ /home/jameskc
Amortization Calculator
Principal, P = 400000
Annual Interest rate (10%), r = 6.5
Number of years, t = 30
Number of payments per year, n = 12
The monthly payment will be p = 2528.27
```

Questions?