**Andrew Wang**

**Homework 3**

1. **Code:**

################

# Author: Andrew Wang

# Date: 9/29/2019

# This program allows the user to input a positive number until a negative number is inputted

# and the program calculates the sum and average of the number inputted

#################

#Allows the user to input a number

number = float(input('Enter a positive number (negative to quit): '))

#Initiallizing total and index

index = 0

total = 0

#While loop that keeps running when the number is positive

while number >= 0:

total = total + number

index = index + 1

number = float(input('Enter a positive number (negative to quit): '))

#Calculation of average

average = total / index

print('Sum = $.2f' %total)

print('Average = %.2f' %average)

**Output:**

**A screenshot of text

Description automatically generated**

1. **Code**

################

# Author: Andrew Wang

# Date: 9/29/2019

# This program allows the user to input total number of years and rainfall amount for each month

# and the program will calculate total number of months, total rainfall, and monthly average rainfall.

#################

#Allows user to input number of years

years = int(input('Enter the number of years: '))

#Initialize index variables

index = 0

year\_count = 1

total = 0

#If statement to determine if total number of years is positive

if years <= 0:

print('Invalid input.')

else:

#While loop to determine if count is less than years

while index < years:

index = index + 1

print('For year No.', year\_count)

Jan = float(input('Enter the rainfall amount for Jan.: '))

Feb = float(input('Enter the rainfall amount for Feb.: '))

Mar = float(input('Enter the rainfall amount for Mar.: '))

Apr = float(input('Enter the rainfall amount for Apr.: '))

May = float(input('Enter the rainfall amount for May.: '))

Jun = float(input('Enter the rainfall amount for Jun.: '))

Jul = float(input('Enter the rainfall amount for Jul.: '))

Aug = float(input('Enter the rainfall amount for Aug.: '))

Sept = float(input('Enter the rainfall amount for Sept.: '))

Oct = float(input('Enter the rainfall amount for Oct.: '))

Nov = float(input('Enter the rainfall amount for Nov.: '))

Dec = float(input('Enter the rainfall amount for Dec.: '))

year\_count = year\_count + 1

total = (Jan + Feb + Mar + Apr + May + Jun + Jul + Aug + Sept + Oct + Nov + Dec ) + total

#Calculates average rainfall

average = total / (12 \* years)

print('There are %d months.' %(12 \* years))

print('The total rainfall is %.2f inches.' %total)

print('The monthly average rainfall is %.2f inches.' %average)

**Output:**

**A close up of text on a white background

Description automatically generated**

1. **Code:**

################

# Author: Andrew Wang

# Date: 9/29/2019

# This program allows the user to input number of organisms, average daily increase, days to multiply

# and the program calculates the population of the organism each day.

#################

#Allows user to input number of organisms

num\_org = float(input('Starting number of organisms: '))

#Allows user to input average daily increase

avg\_inc = float(input('Average daily increae, in percent: '))

#Allows user to input number of days to multiply

days\_mult = float(input('Number of days to multiply: '))

#Initializing index

index = 0

days = 0

print('Day Approximate\tPopulation')

#While loop to determine index exceedes days to multiply

while index < days\_mult:

days = days + 1

index = index + 1

print('%d\t\t\t%.4f' %(days, num\_org))

#Calculates the number of organisms each day

num\_org = num\_org \* (avg\_inc / 100 + 1)

**Output:**

**A screenshot of a social media post

Description automatically generated**

1. **Code:**

################

# Author: Andrew Wang

# Date: 9/29/2019

# This program prints out 7 \* initially and decrease number of \* each line until there is 1 \* left

#################

#Initialize index

index = 7

#While loop to determine if index is larger than 0

while index > 0:

#Intialize count

count = 0

#While loop to print out \* corresponding with number of index

while index > count:

print('\*', end='')

count = count + 1

print(' ')

index = index – 1

**Output:**

**A screenshot of a cell phone

Description automatically generated**

1. **Code:**

################

# Author: Andrew Wang

# Date: 9/29/2019

# This program uses turtle graphs and draws 5 different colored circle at different locations

#################

#Initializes turtle graphics and draws blue circle

import turtle

turtle.setup(2000, 2000)

turtle.pensize(10)

turtle.pencolor('blue')

turtle.circle(50)

#Draws yellow circle

turtle.penup()

turtle.goto(62.5,-50)

turtle.pendown()

turtle.pencolor('yellow')

turtle.circle(50)

#Draws black circle

turtle.penup()

turtle.goto(125,0)

turtle.pendown()

turtle.pencolor('black')

turtle.circle(50)

#Draws green circle

turtle.penup()

turtle.goto(187.5,-50)

turtle.pendown()

turtle.pencolor('green')

turtle.circle(50)

#Draws red circle and terminates turtle

turtle.penup()

turtle.goto(250,0)

turtle.pendown()

turtle.pencolor('red')

turtle.circle(50)

turtle.done()

**Output:**

**A close up of a logo

Description automatically generated**

1. **Code:**

################

# Author: Andrew Wang

# Date: 9/29/2019

# This program allows the user to input number of lines and the program prints out

# "##" with space in between them as each line progresses

#################

#Allows the user to input number of lines

lines = int(input('Enter the number of lines: '))

print ("##")

#For loop that increase a space between as each line progresses

for index in range(lines - 1):

print ("#",index \* " " + "#")

**Output:**

A screenshot of a cell phone

Description automatically generated