Lab 5P: Repetition Structures_ 5P_RSX

GADDIS Text Pages 199 - 201

Using the IDLE editor (FILE/New Window)……

Name: ____________Jeff Couch_____________

EXTRA CREDIT (205 Pts)

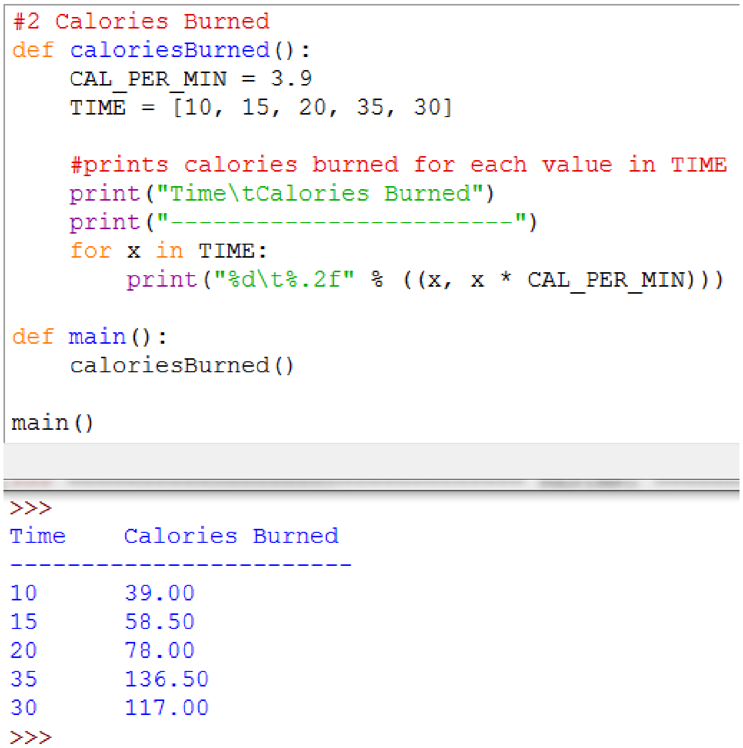
For EACH EXTRA CREDIT Project you complete:

Take one Screenshot of both the EDIT window disiplaying your program and the SHELL window displaying your program results and paste where indicated.

SAVE your program. All extra credit programs must be submitted with this lab template

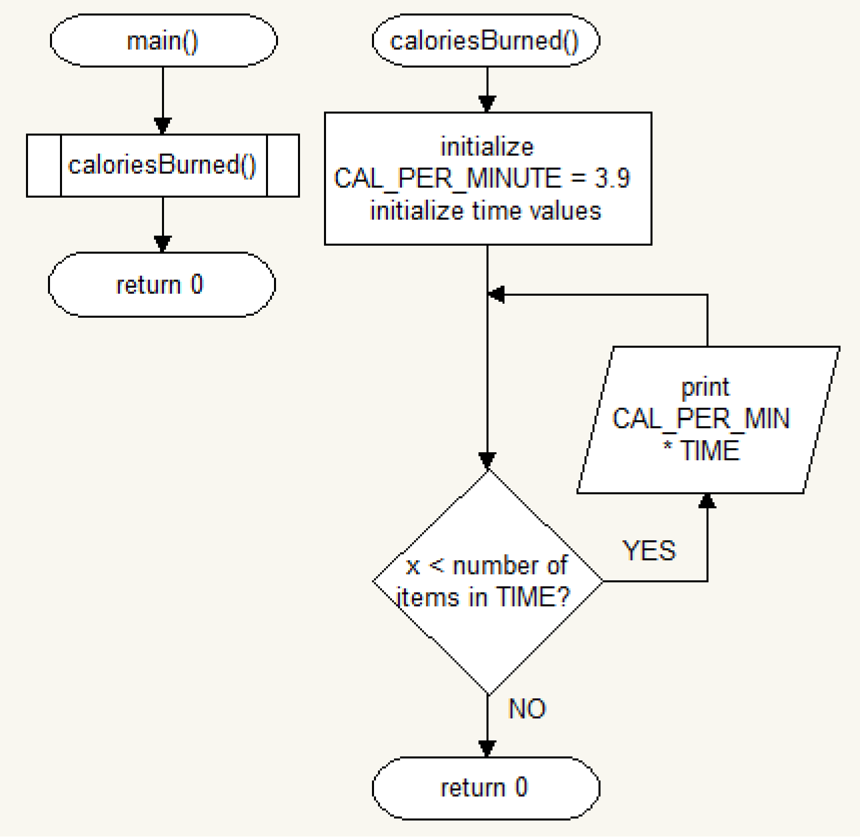
Project 2. Calories Burned (25 Pts)

Input: Minutes = 10,15,20,25,30



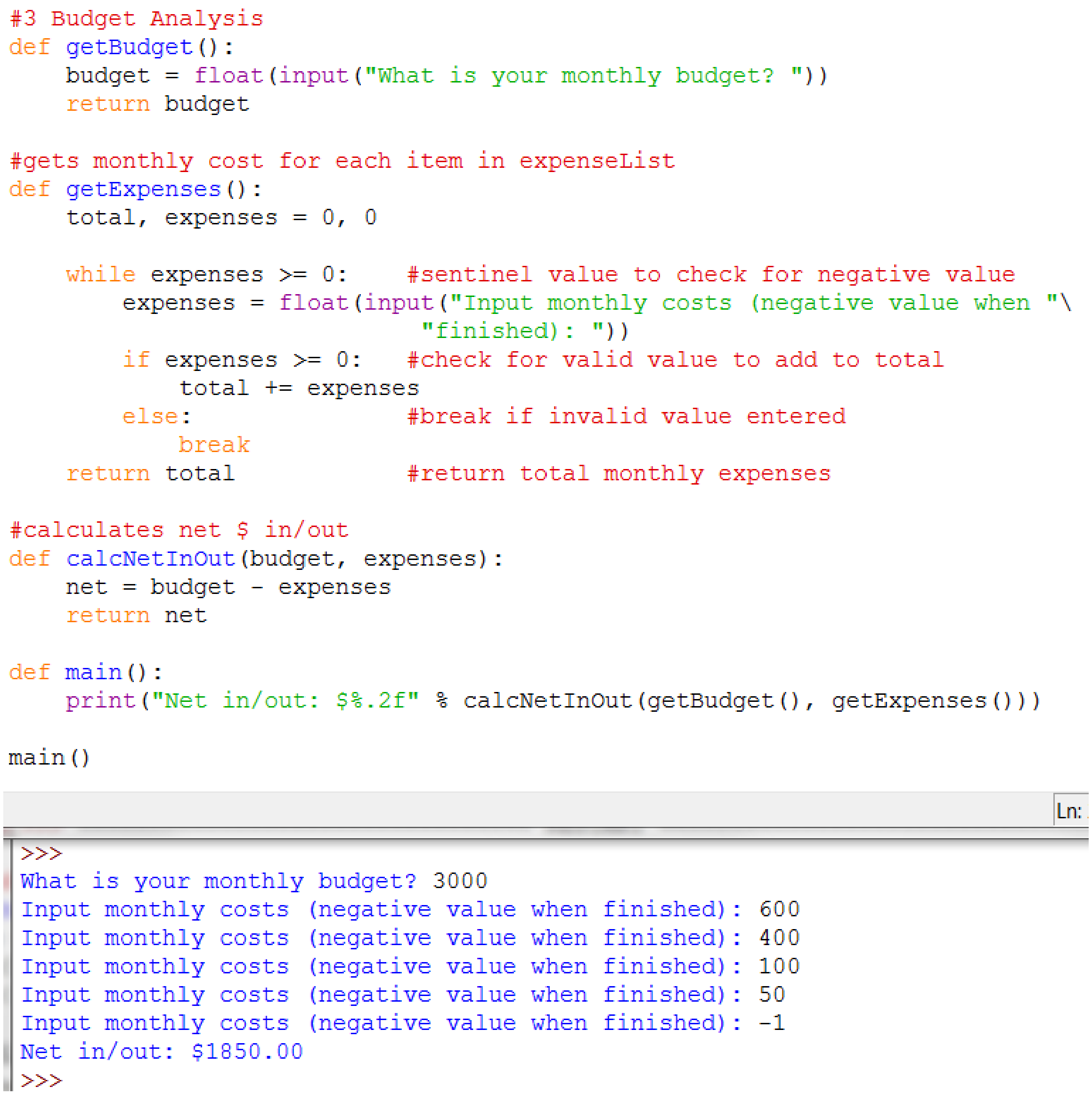
```
#2 Calories Burned
def caloriesBurned():
    CAL_PER_MIN = 3.9
    TIME = [10, 15, 20, 35, 30]

    #prints calories burned for each value in TIME
    print("Time\tCalories Burned")
    print("------------------------")
    for x in TIME:
        print("%d\t%.2f" % ((x, x * CAL_PER_MIN)))

def main():
    caloriesBurned()

main()
```

```
>>> 
Time    Calories Burned
------------------------
10      39.00
15      58.50
20      78.00
35      136.50
30      117.00
>>> 
```



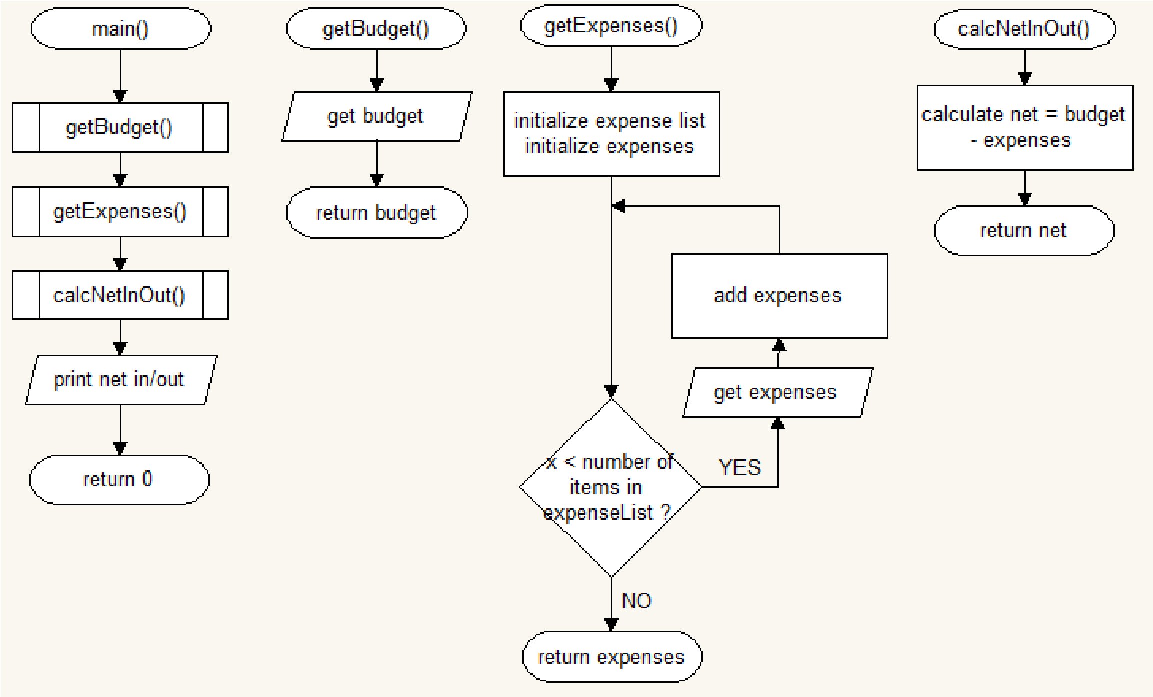
Project 3: Budget Analysis (25 Pts)

Input: **Budget($) = 2500**



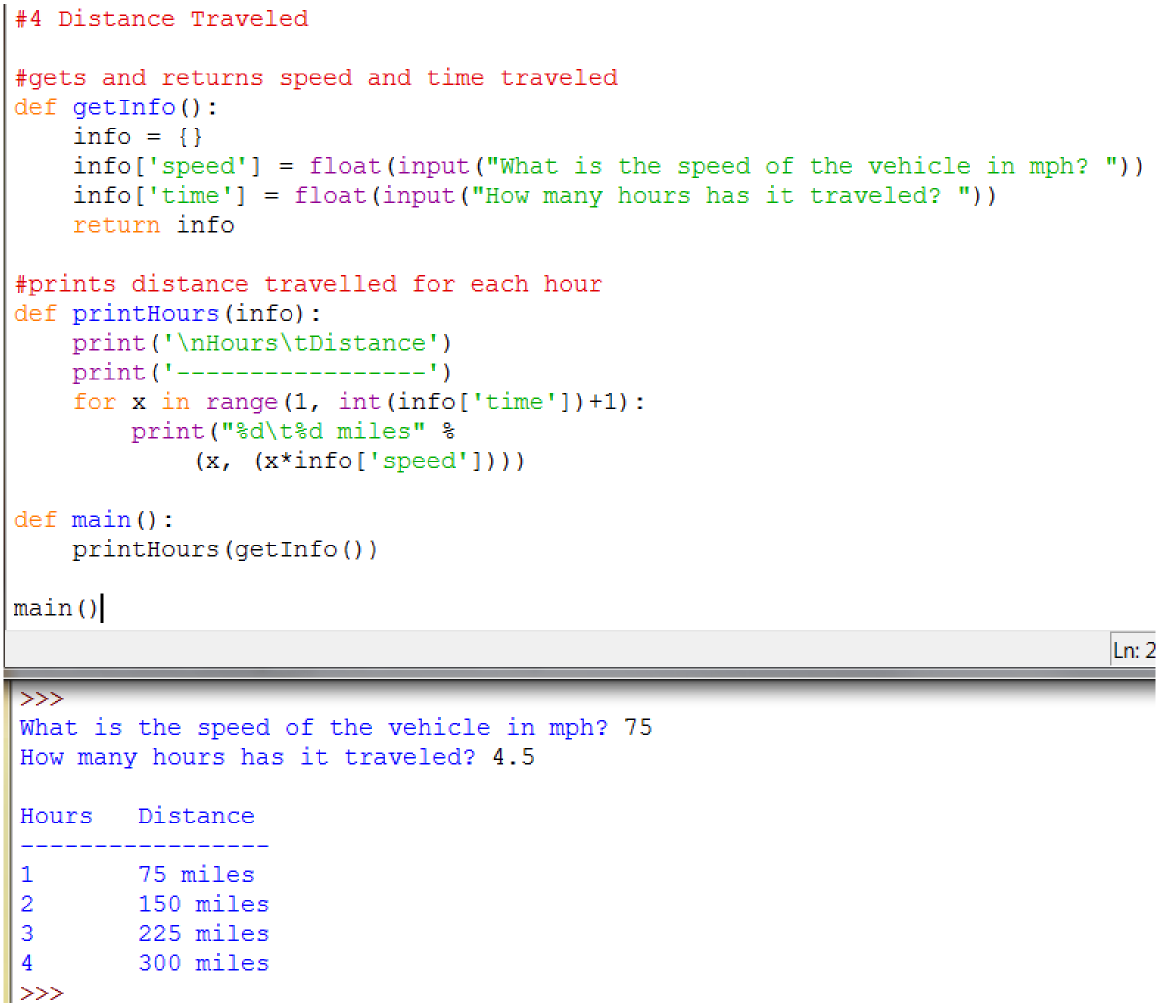
```
#3 Budget Analysis
def getBudget():
    budget = float(input("What is your monthly budget? "))
    return budget

#gets monthly cost for each item in expenseList
def getExpenses():
    total, expenses = 0, 0

    while expenses >= 0:    #sentinel value to check for negative value
        expenses = float(input("Input monthly costs (negative value when "\
                                    "finished): "))
        if expenses >= 0:   #check for valid value to add to total
            total += expenses
        else:               #break if invalid value entered
            break
    return total            #return total monthly expenses

#calculates net $ in/out
def calcNetInOut(budget, expenses):
    net = budget - expenses
    return net

def main():
    print("Net in/out: $%.2f" % calcNetInOut(getBudget(), getExpenses()))

main()
```

```
>>> 
What is your monthly budget? 3000
Input monthly costs (negative value when finished): 600
Input monthly costs (negative value when finished): 400
Input monthly costs (negative value when finished): 100
Input monthly costs (negative value when finished): 50
Input monthly costs (negative value when finished): -1
Net in/out: $1850.00
>>> 
```



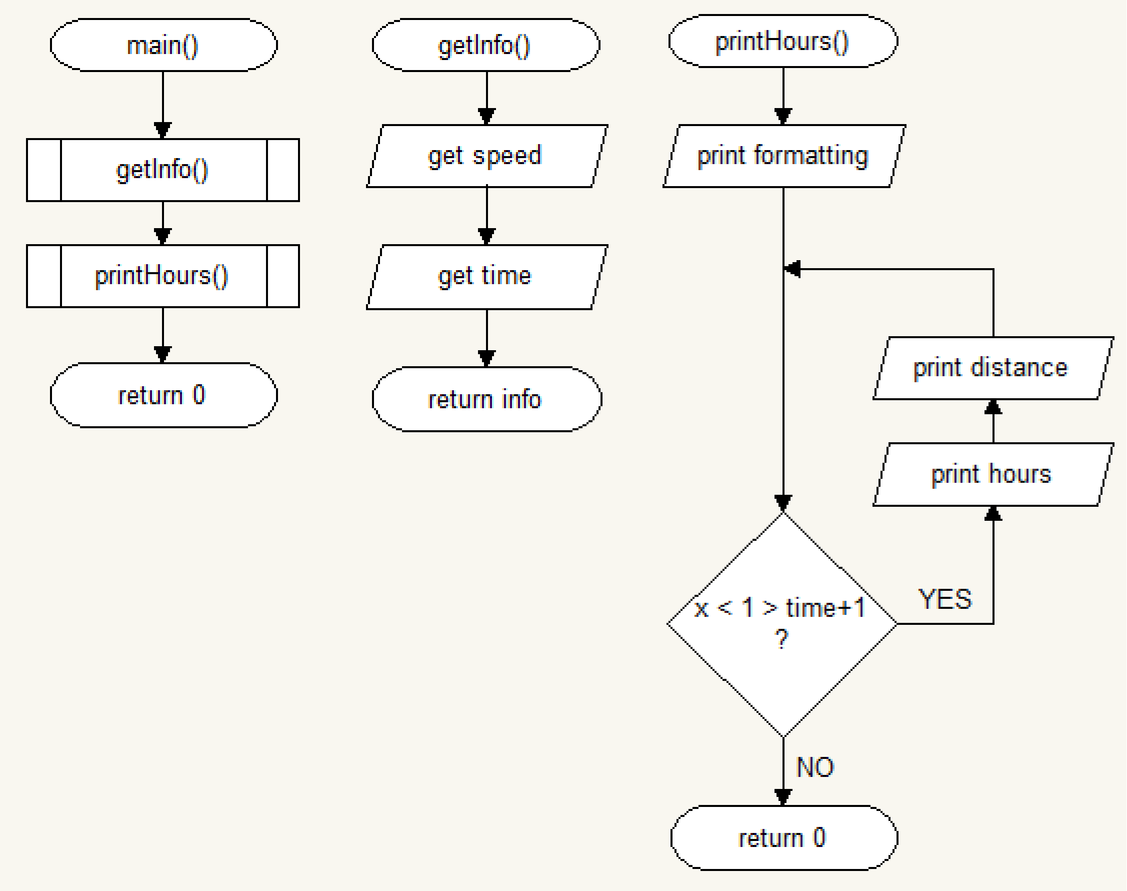
Project 4: Distance Traveled (25 Pts)

Input: Speed(mph) = 75, Time(hrs) = 4.5



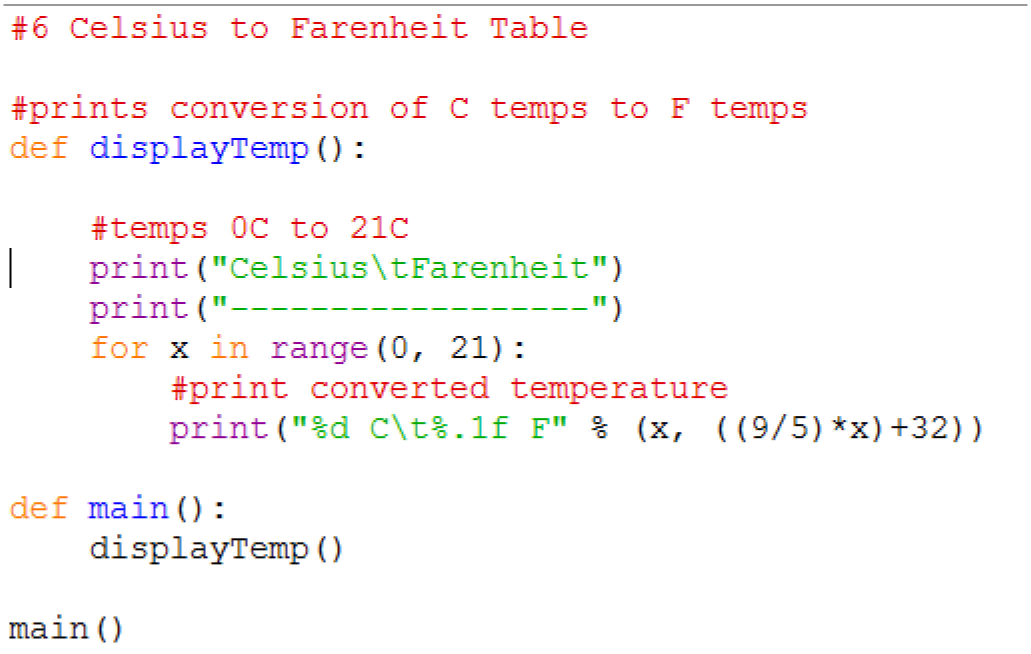
```
#4 Distance Traveled

#gets and returns speed and time traveled
def getInfo():
    info = {}
    info['speed'] = float(input("What is the speed of the vehicle in mph? "))
    info['time'] = float(input("How many hours has it traveled? "))
    return info

#prints distance travelled for each hour
def printHours(info):
    print('\nHours\tDistance')
    print('-----------------')
    for x in range(1, int(info['time'])+1):
        print("%d\t%d miles" %
             (x, (x*info['speed'])))

def main():
    printHours(getInfo())

main()
```

```
>>>
What is the speed of the vehicle in mph? 75
How many hours has it traveled? 4.5

Hours   Distance
-----------------
1       75 miles
2       150 miles
3       225 miles
4       300 miles
>>>
```

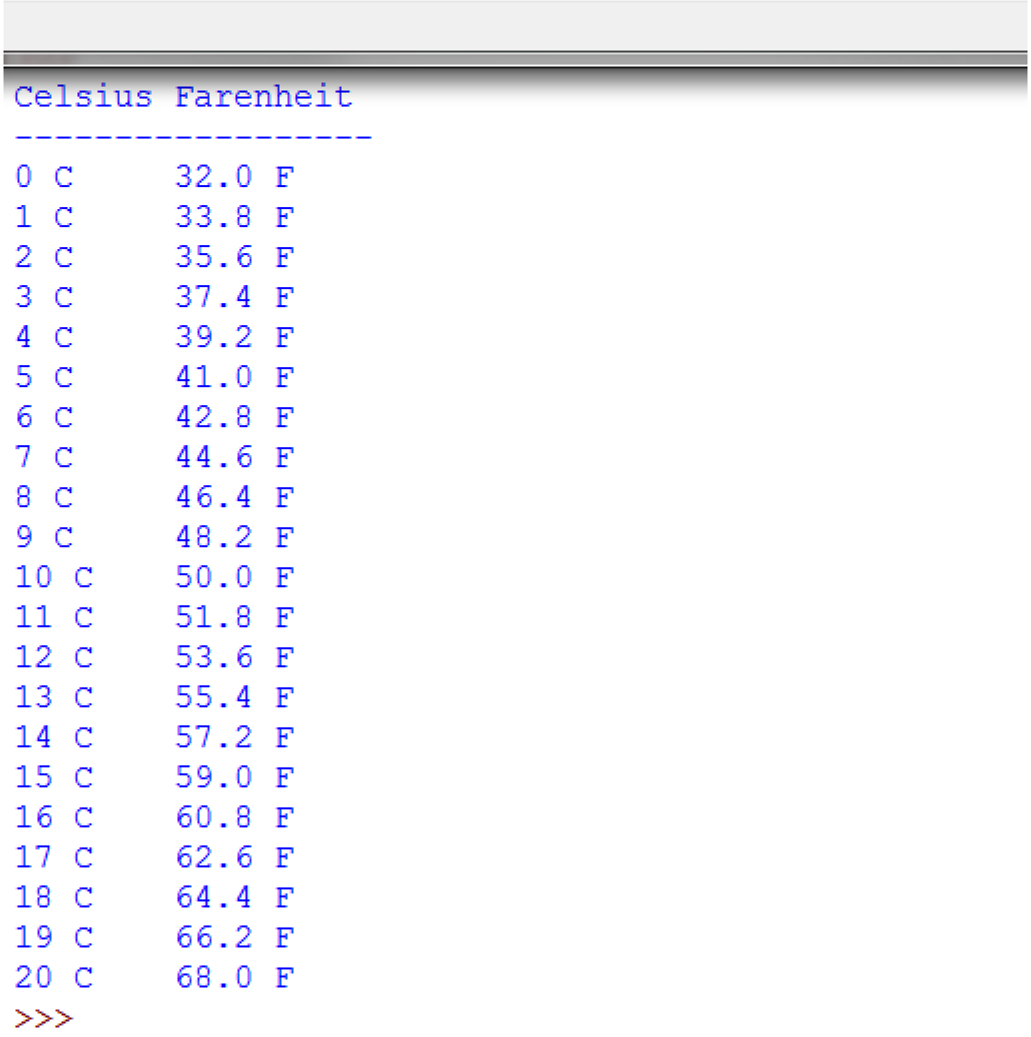


Project 6: Celsius to Fahrenheit Table (25 Pts)

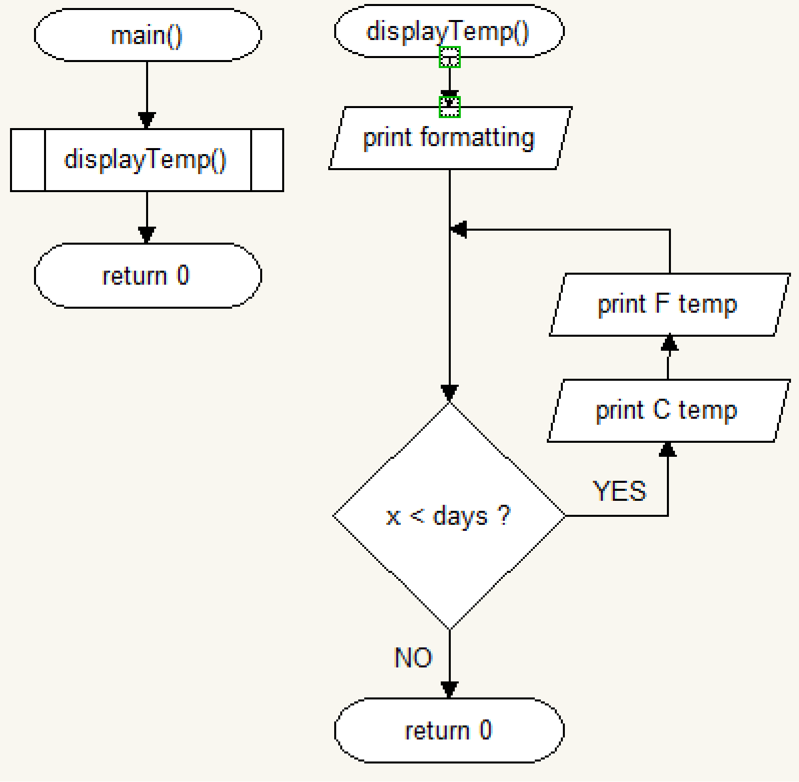
Input: Temps(C) = 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20



```
#6 Celsius to Farenheit Table

#prints conversion of C temps to F temps
def displayTemp():

    #temps 0C to 21C
    print("Celsius\tFarenheit")
    print("------------------")
    for x in range(0, 21):
        #print converted temperature
        print("%d C\t%.1f F" % (x, ((9/5)*x)+32))

def main():
    displayTemp()

main()
```

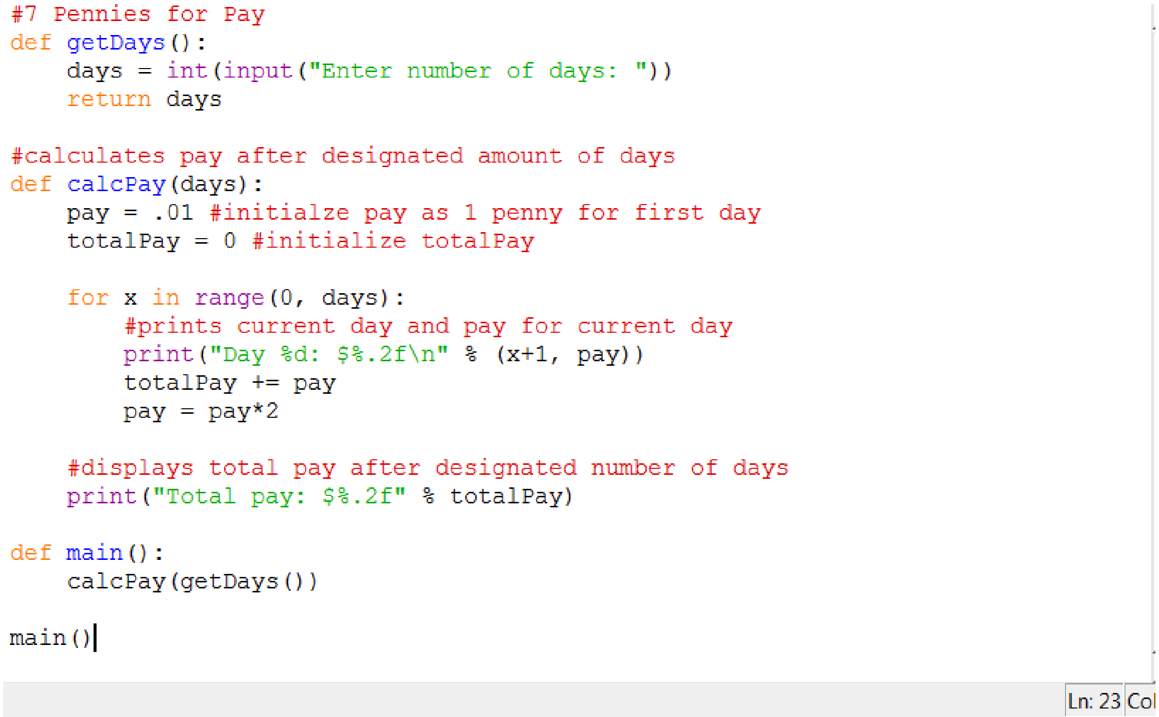


```
Celsius Farenheit
------------------
0 C     32.0 F
1 C     33.8 F
2 C     35.6 F
3 C     37.4 F
4 C     39.2 F
5 C     41.0 F
6 C     42.8 F
7 C     44.6 F
8 C     46.4 F
9 C     48.2 F
10 C    50.0 F
11 C    51.8 F
12 C    53.6 F
13 C    55.4 F
14 C    57.2 F
15 C    59.0 F
16 C    60.8 F
17 C    62.6 F
18 C    64.4 F
19 C    66.2 F
20 C    68.0 F
>>>
```

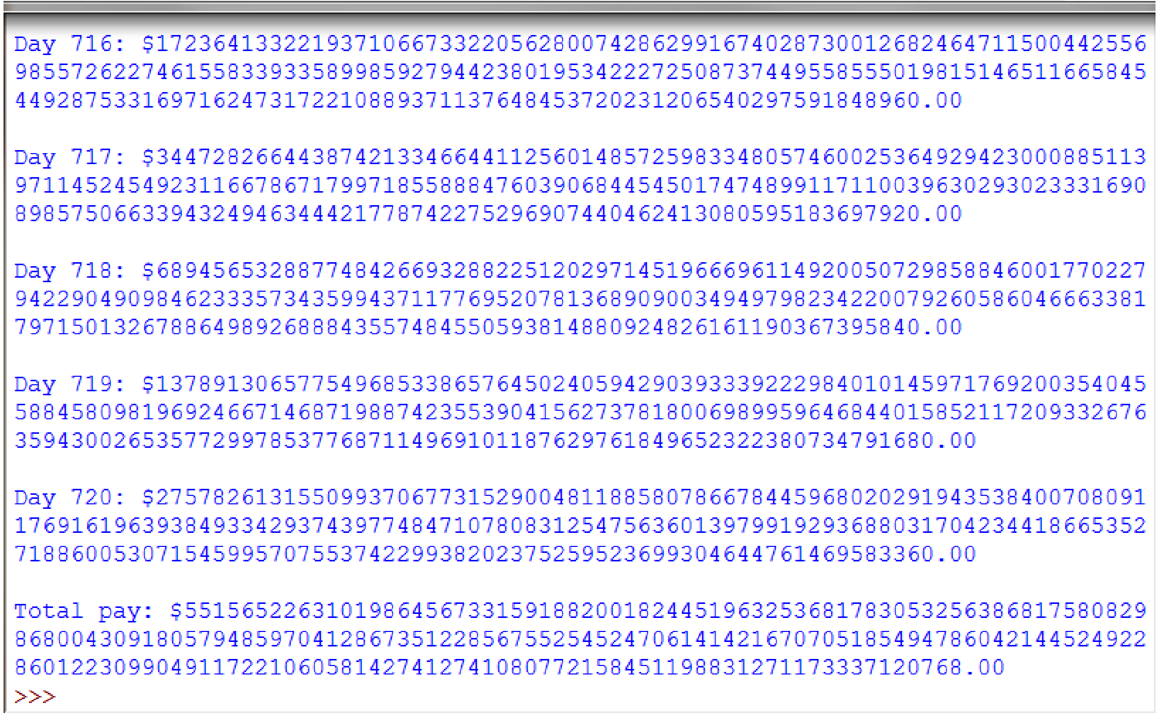


Project 7: Pennies for Pay (25 Pts)

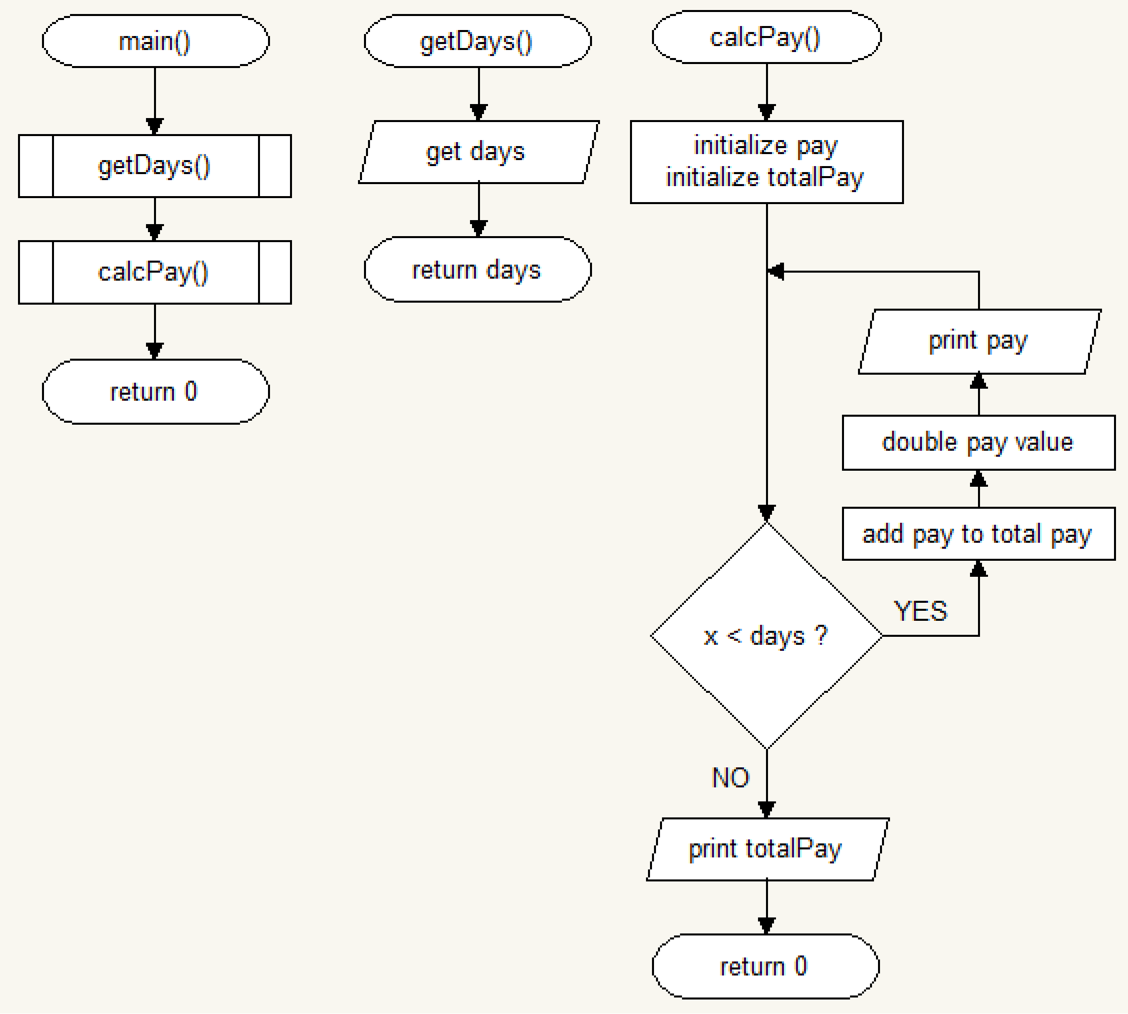
Input: Number of days = 720



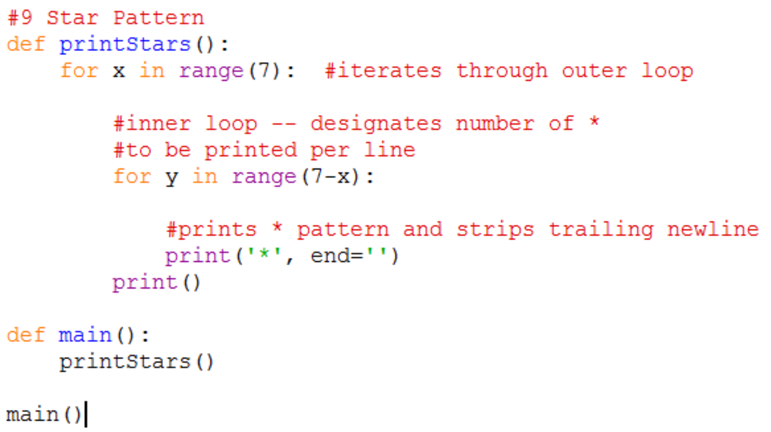
```
#7 Pennies for Pay
def getDays():
    days = int(input("Enter number of days: "))
    return days

#calculates pay after designated amount of days
def calcPay(days):
    pay = .01 #initialze pay as 1 penny for first day
    totalPay = 0 #initialize totalPay

    for x in range(0, days):
        #prints current day and pay for current day
        print("Day %d: $%.2f\n" % (x+1, pay))
        totalPay += pay
        pay = pay*2

    #displays total pay after designated number of days
    print("Total pay: $%.2f" % totalPay)

def main():
    calcPay(getDays())

main()
```



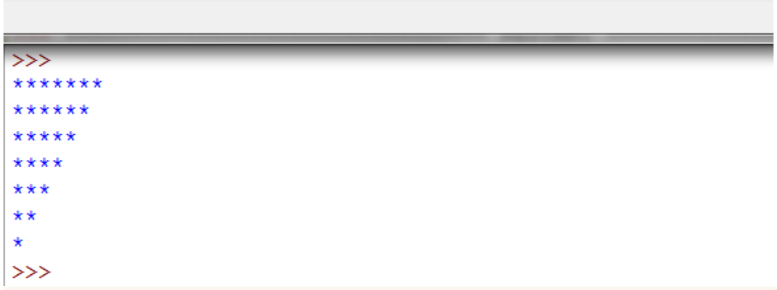
```
Day 716: $1723641332219371066733220562800742862991674028730012682464711500442556
98557262274615583393358998592794423801953422272508737449558555019815146511665845
4492875331697162473172210889371137648453720231206540297591848960.00

Day 717: $3447282664438742133466441125601485725983348057460025364929423000885113
97114524549231166786717997185588847603906844545017474899117110039630293023331690
8985750663394324946344421778742275296907440462413080595183697920.00

Day 718: $6894565328877484266932882251202971451966696114920050729858846001770227
94229049098462333573435994371177695207813689090034949798234220079260586046663381
7971501326788649892688843557484550593814880924826161190367395840.00

Day 719: $1378913065775496853386576450240594290393339222984010145971769200354045
58845809819692466714687198874235539041562737818006989959646844015852117209332676
35943002653577299785377687114969101187629761849652322380734791680.00

Day 720: $2757826131550993706773152900481188580786678445968020291943538400708091
17691619639384933429374397748471078083125475636013979919293688031704234418665352
71886005307154599570755374229938202375259523699304644761469583360.00

Total pay: $55156522631019864567331591882001824451963253681783053256386817580829
86800430918057948597041286735122856755254524706141421670705185494786042144524922
8601223099049117221060581427412741080772158451198831271173337120768.00
>>>
```



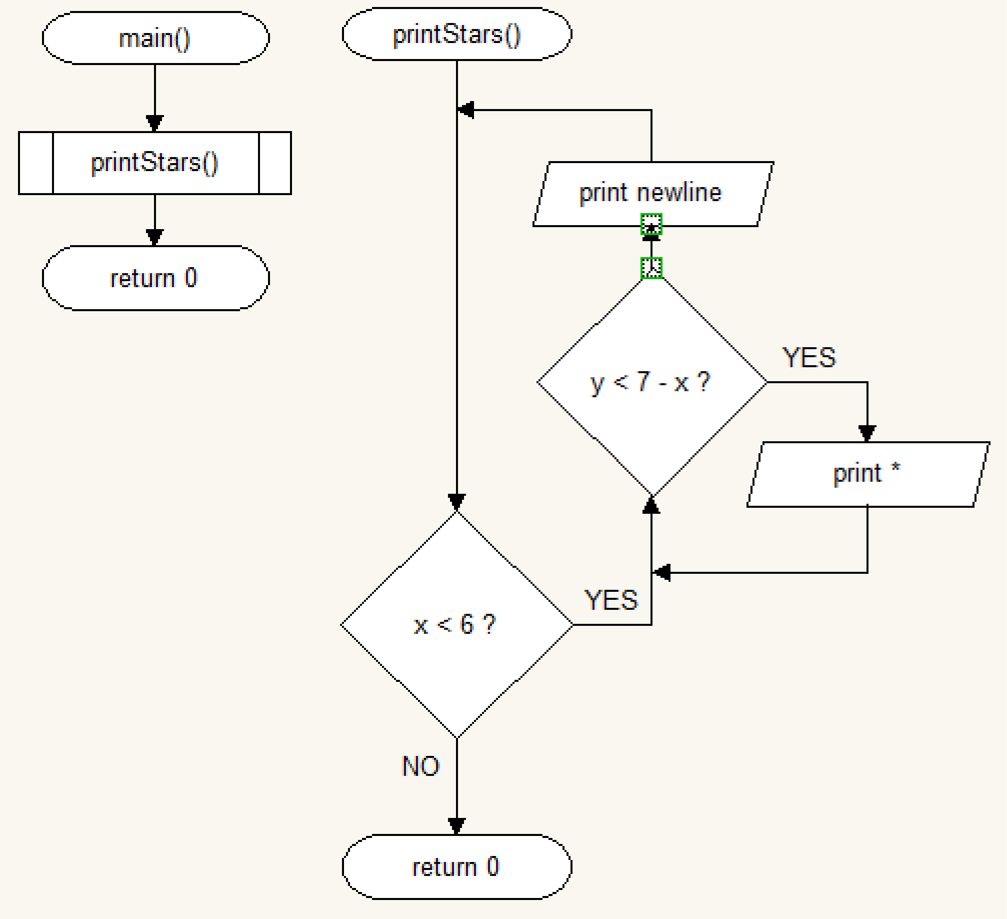
Project 9: Write a program that uses nested loops to draw this pattern: **(35 Pts)**



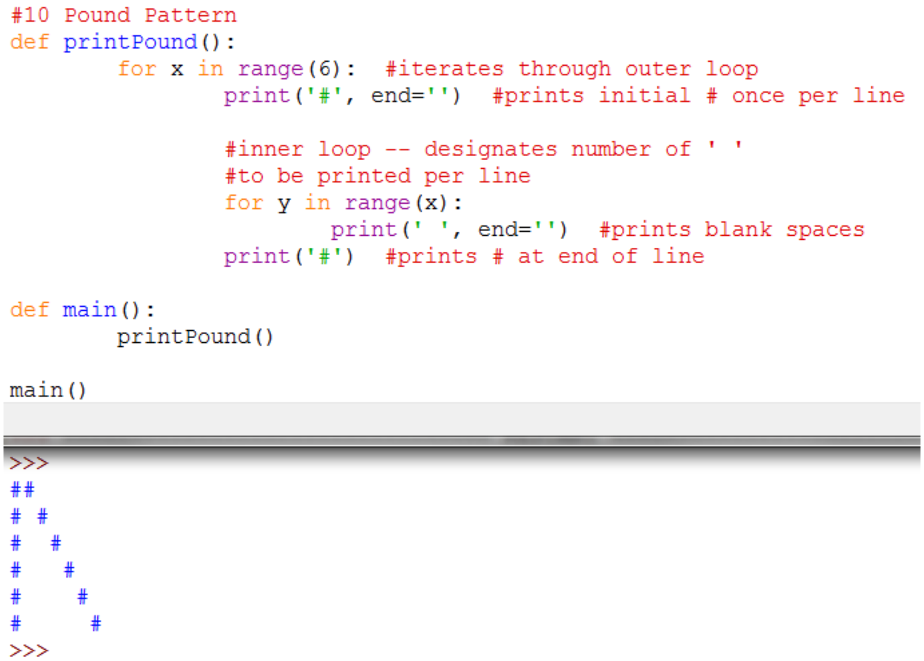
```
#9 Star Pattern
def printStars():
    for x in range(7):  #iterates through outer loop

        #inner loop -- designates number of *
        #to be printed per line
        for y in range(7-x):

            #prints * pattern and strips trailing newline
            print('*', end='')
        print()

def main():
    printStars()

main()
```



```
>>>
*******
******
*****
****
***
**
*
>>>
```

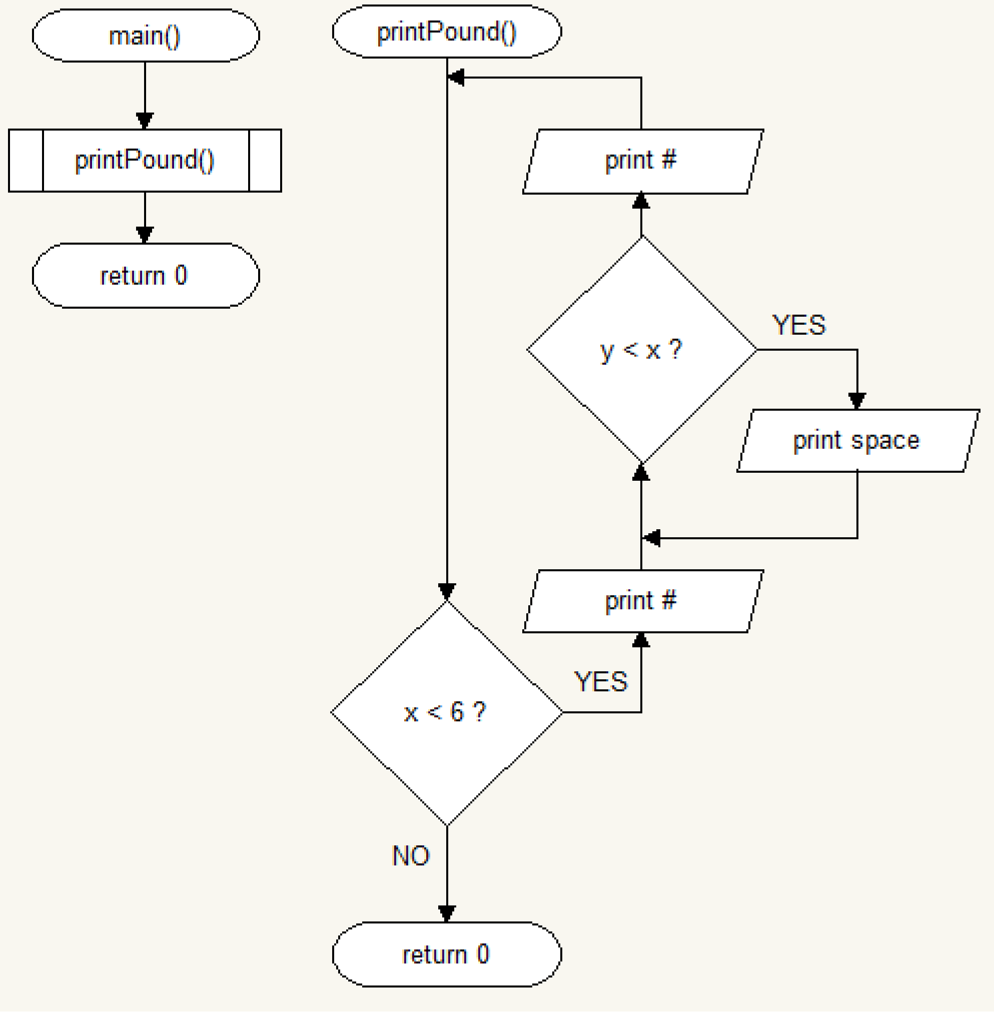


Project 10: Write a program that uses nested loops to draw this pattern: (45 Pts)



```
#10 Pound Pattern
def printPound():
        for x in range(6):  #iterates through outer loop
                print('#', end='')  #prints initial # once per line

                #inner loop -- designates number of ' '
                #to be printed per line
                for y in range(x):
                        print(' ', end='')  #prints blank spaces
                print('#')  #prints # at end of line

def main():
        printPound()

main()
```

```
>>>
##
# #
#  #
#   #
#    #
#     #
>>>
```



main()
printPound()
return 0
printPound()
print #
y < x ?
YES
print space
print #
YES
x < 6 ?
NO
return 0