# CIS 1400 NET05 – Fall 2020 Week 12 Homework

# Rooftop Gardens at COD Revisited using Files

Write the Pseudocode, Flowchart, and Python code for the following programming problem from your week 10 homework. Pseudocode for main, and file processing modules is provided below. Compare it to the original pseudocode from week 10, and figure out where it belongs. Do this first, before attempting your Flowchart!

Note that in addition to what the program already does, it should create a file called savings.txt and store the savings array to a file. This should be done in append mode in Python.

Be sure to upload your flowchart and WORKING & FULLY TESTED Python program to Blackboard. Program MUST contain a header and good level of comments and documentation, which makes up 25% of the program grade.

What if COD decided to implemented rooftop gardens as a way to promote energy efficiency and save money? Let’s write a program that will allow the user to enter the energy bills from January to December for the year prior to going green (prior to implementing the rooftop gardens). Next, allow the user to enter the energy bills from January to December of the past year after implementing the rooftop gardens, or going green. The program should calculate the energy difference from the two years and display the two years’ worth of data, along with the savings. Additionally, the savings array should be printed to a file called savings.txt.

## The Pseudocode

Module main()

//Declare local variables

Declare endProgram = “no”

Declare Real notGreenCost[12]

Declare Real goneGreenCost[12]

Declare Real savings[12]

Declare String months[12] = “January”, “February”, “March”, “April”, “May”, “June”,

“July”, “August”, “September”, “October”, “November”, “December” Declare Integer option = 0

While endProgram == “no” //function calls If option == 1 Then

getNotGreen(notGreenCost, months)

1



### CIS 1400 NET05 – Fall 2020 Week 12 Homework

getGoneGreen(goneGreenCost, months) energySaved(notGreenCost, goneGreenCosts, savings) Else If option == 2 Then

displayInfo(notGreenCost, goneGreenCosts, savings, months) Else If option == 3 Then writeToFile(months, savings) Else If option == 4 Then

readFromFile(months, savings)

End If

Display “Do you want to end the program? Yes or no”

Input endProgram

End While

End Module

Module writeToFile(String months[], Real savings[])

Declare outFile AppendMode savingsFile

Open savingsFile “savings1.txt”

Write savingsFile “Savings”

Declare Integer counter = 0

While counter < 12

Write savingsFile months[counter]

Write savingsFile savings[counter]

Set counter = counter + 1

End While

 Close savingsFile

End Module

Module readFromFile(String months[], Real savings[])

Declare inFile savingsFile Open inFile “savings1.txt”

Read savingsFile str1

Display str1

Read savingsFile months

Display months

Read savingsFile savings

Display savings

Close inFile

End Module

The Flowchart





## The Python Code

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

# Name: David White

# Class: CIS-1400

# Assignment: Lab 10-4

# File: lab10-4.py

# Purpose: energy efficiency savings

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

print('\n\*\*\*David White\*\*\*\n') # Display author's name

# main function

def main():

# declare variables and arrays

# loop control var

endProgram = "no"

# array for costs before going green

notGreencost = [0] \* 12

# array for costs after going green

goneGreencost = [0] \* 12

# array for savings

savings = [0] \* 12

months = ['January', 'February', 'March', 'April', 'May', 'June', 'July',

'August', 'September', 'October', 'November', 'December']

# decision control

option = int(0)

while endProgram == "no":

# ask user for input

print('Enter 1 to enter data')

print("Enter 2 to show info")

print("Enter 3 to write to savings1.txt")

print("Enter 4 to show savings1.txt")

option = int(input('Enter number: '))

if option == 1:

# user enters data before going green

notGreencost = getNotgreen(notGreencost, months)

# user enters data after going green

goneGreencost = getGonegreen(goneGreencost, months)

# calculate savings

savings = energySaved(notGreencost, goneGreencost, savings)

# end if

elif option == 2:

printInfo(notGreencost, goneGreencost, savings, months)

# end if

elif option == 3:

writeToFile(months, savings)

print('Data written to savings1.txt')

print()

# end if

elif option == 4:

readFromFile(months, savings)

print()

# end if

endProgram = input("Do you want to end program? Enter yes or no: ")

print()

while not (endProgram == 'yes' or endProgram == 'no'):

print('Please enter yes or no')

endProgram = input('Do you want to end program? (Enter no or yes): ')

print()

# get cost before going green

def getNotgreen(notGreencost, months):

# reset counter

counter = 0

while counter < len(months):

print("Enter NOT GREEN energy costs for", months[counter])

notGreencost[counter] = input("--> ")

# increment counter

counter = counter + 1

# extra line for separation

print()

# end while

return notGreencost

# get cost after going green

def getGonegreen(goneGreencost, months):

# reset counter

counter = 0

while counter < len(months):

print("Enter GONE GREEN energy costs for", months[counter])

goneGreencost[counter] = input("--> ")

# increment counter

counter = counter + 1

# extra line for separation

print()

# end while

return goneGreencost

# calculate savings

def energySaved(notGreencost, goneGreencost, savings):

# reset counter

counter = 0

while counter < len(savings):

savings[counter] = float(notGreencost[counter]) - float(goneGreencost[counter])

# increment counter

counter = counter + 1

# end while

return savings

def writeToFile(months, savings):

# reset counter

counter = int(0)

# open savings1.txt in append mode

outFile = open("savings1.txt", "a")

# write "savings" in savings.txt

outFile.write("Savings\n")

# write each months savings to savings1.txt

while counter < 12:

outFile.write(months[counter] + '\n')

outFile.write(str(savings[counter]) + '\n')

# increment counter

counter = counter + 1

# end while

# close savings.txt

outFile.close()

def readFromFile(months, savings):

# open savings1.txt in read mode

inFile = open("savings1.txt", "r")

str1 = inFile.read()

print(str1)

months = inFile.read()

print(months)

savings = inFile.read()

print(savings)

inFile.close()

# show info

def printInfo(notGreencost, goneGreencost, savings, months):

# reset counter

counter = 0

while counter < 12:

# print extra line for separation

print()

# displays month

print("Information for", months[counter])

# displays savings for current month

print("Savings: $", savings[counter])

# displays cost before going green

print("Not Green Costs: $", notGreencost[counter])

# displays cost after going green

print("Gone Green Costs: $", goneGreencost[counter])

# extra line for separation

print()

# increment counter

counter = counter + 1

# end while

# call main

main()