CMPE 365 Lab 4: Greedy Algorithms

Cache Angus

20000629

October 12, 2018

import csv  
import random  
  
#get in the csv values and put them into an x and y value  
def \_importstart\_csv(filestart):  
 start = []  
  
 #get the values from the csv file  
 with open(filestart, 'rt', encoding='utf-8') as csvfile:  
 startimport = csv.reader(csvfile, delimiter='\n')  
 for row in startimport:  
 num = float(row[0])  
 start.append(num)  
 return start  
def \_importfinish\_csv(fileend):  
 finish = []  
 with open(fileend, 'rt', encoding='utf-8') as csvfile:  
 finishimport = csv.reader(csvfile, delimiter='\n')  
 for row in finishimport:  
 num = float(row[0])  
 finish.append(num)  
 return finish  
def \_delay(list, maxDelay):  
 index = random.randint(0,len(list)) #generate a random number of planes to be delayed  
 for i in range(index): #iterate through the arrival and departure times of each plane and add a random delay  
 delay = random.uniform(0, maxDelay)  
 list[i] += delay  
 return list  
def \_sort\_(list):  
 if (len(list) <= 1):  
 return list  
 #sort the list by using the finish time  
 list.sort(key=lambda x: x[0])  
  
 return list  
  
#when doing this stuff try to edit it so that it checks against all of the ones that have already been checked  
  
def \_schedulealgorithm\_(list):  
 checked = []  
 #put the first to finish into the list of checked values  
 checked.append(list[0])  
 #gate starts at 1 because we know the first flight has to take off  
 gate = 1  
 #maybe i should use a while loop  
 for i in range(1, len(list)):  
 val = len(checked)  
 #check all of those that are currently in checked and therefore have been or are at a gate  
 for t in range(0, val):  
 #if the start of the next one is before the end of the one currently running  
 if list[i][0] > checked[t][1]:  
 del checked[t]  
 gate -= 1  
 break  
 gate += 1  
  
 #add the newly checked one to the checked list  
 checked.append(list[i])  
 #after checking through all of the flight times, return the number of gates that needed to be added  
 return gate  
  
  
starting = \_importstart\_csv('start2.csv')  
ending = \_importfinish\_csv('finish2.csv')  
  
starting = \_delay(starting, 1.0)  
ending = \_delay(ending, 1.0)  
flight = []  
  
for i in range(0, len(starting)):  
 times = [starting[i], ending[i]]  
 flight.append(times)  
  
sorted\_flight = \_sort\_(flight)  
  
gates\_needed = \_schedulealgorithm\_(sorted\_flight)  
print(gates\_needed)