İSTANBUL ŞEHİR ÜNİVERSİTY

ENGR 212: Programming Practice

Week 13

Optimization

İSTANBUL ŞEHİR ÜNİVERSİTY

- Optimization finds the best solution to a problem by trying many different solutions and scoring them to determine their quality.
- Optimization is typically used in cases where there are too many possible solutions to try them all.
- The simplest but least effective method of searching for solutions is just trying a few thousand random guesses and seeing which one is best.

Group Travel Planning

İSTANBUL ŞEHİR ÜNİVERSİTY

- Have you ever planned a trip for a group of people?
- You must have thought about total cost, time spent waiting at airports, and time taken off work.

Group Travel Planning

İSTANBUL ŞEHİR ÜNİVERSİTY

- Let's plan a trip for a group of people (the Glass family in this example) from different locations all arriving at NewYork, La Guardia Airport.

Group Travel Planning

İSTANBUL ŞEHİR UNIVERSITY

Group Travel Planning

İSTANBUL ŞEHİR ÜNİVERSİTY

```
import time
import random
import math

people = [('Seymour','BOS'),
          ('Franny','DAL'),
          ('Zooey','CAK'),
          ('Walt','MIA'),
          ('Buddy','ORD'),
          ('Les','OMA')]

# LaGuardia airport in New York
destination='LGA'
```

Group Travel Planning

İSTANBUL ŞEHİR ÜNİVERSİTY

- schedule.txt
  - This file contains origin, destination, departure time, arrival time, and price for a set of flights

```
LGA,MIA,20:27,23:42,169
MIA,LGA,19:53,22:21,173
LGA,BOS,6:39,8:09,86
BOS,LGA,6:17,8:26,89
LGA,BOS,8:23,10:28,149
```

Group Travel Planning

İSTANBUL ŞEHİR ÜNİVERSİTY

- Let's prepare a flights dictionary:

```
flights = {}
for line in file('schedule.txt'):
    origin, dest, depart, arrive, price = line.strip().split(',')
    flights.setdefault((origin, dest), [])

    # Add details to the list of possible flights
    flights[(origin, dest)].append((depart, arrive, int(price)))
```

Representing Solutions

İSTANBUL ŞEHİR ÜNİVERSİTY

- There may be different alternatives.
- Should be generic enough so that any optimization algorithm may be used.
- List of numbers is a common representation.
  - Each number represents the flight a person takes

Example: `[1,4,3,2,7,3,6,3,2,4,5,3]`

Printing a Solution

İSTANBUL ŞEHİR ÜNİVERSİTY

```
def printschedule(r):
    print r
    for d in range(len(r)/2):
        name = people[d][0]
        origin = people[d][1]
        out = flights[(origin,destination)][r[2*d]]
        ret = flights[(destination,origin)][r[2*d+1]]
        print '%10s%10s %5s-%5s $%3s %5s-%5s $%3s' % (name,origin,
                                                      out[0],out[1],out[2],
                                                      ret[0],ret[1],ret[2])
```

Printing a Solution

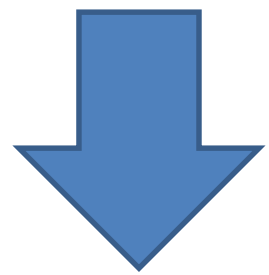
İSTANBUL ŞEHİR ÜNİVERSİTY

```
import optimization
s=[1, 4, 3, 2, 7, 3, 6, 3, 2, 4, 5, 3]
optimization.printschedule(s)
```

Printing a Solution

İSTANBUL ŞEHİR ÜNİVERSİTY

```
import optimization
s=[1, 4, 3, 2, 7, 3, 6, 3, 2, 4, 5, 3]
optimization.printschedule(s)
```



```
[1, 4, 3, 2, 7, 3, 6, 3, 2, 4, 5, 3]
   Seymour       BOS  8:04-10:11 $ 95 12:08-14:05 $142
    Franny       DAL 12:19-15:25 $342 10:51-14:16 $256
     Zooey       CAK 10:53-13:36 $189  9:58-12:56 $249
      Walt       MIA  9:15-12:29 $225 16:50-19:26 $304
     Buddy       ORD 16:43-19:00 $246 10:33-13:11 $132
       Les       OMA 11:08-13:07 $175 15:07-17:21 $129
```

The cost function

İSTANBUL ŞEHİR ÜNİVERSİTY

- **Price**: The total price of all the plane tickets, or possibly a weighted average.
- **Travel time**: The total time that everyone has to spend on a plane.
- **Waiting time**: Time spent at the airport waiting for the other members of the party to arrive.

The cost function

İSTANBUL ŞEHİR ÜNİVERSİTY

- **Departure time**: Flights that leave too early in the morning may impose an additional cost by requiring travelers to miss out on sleep.
- **Car rental period**: If the party rents a car, they must return it earlier in the day than when they rented it, or be forced to pay for a whole extra day.

The cost function

İSTANBUL ŞEHİR ÜNİVERSİTY

- After choosing some variables that impose costs, you'll need to determine how to combine them into a single number.
- How much money that time on the plane or time waiting in the airport is worth?

The cost function

İSTANBUL ŞEHİR ÜNİVERSİTY

- You might decide that:
  - it's worth spending $1 for every minute saved on air travel (this translates into spending an extra $90 for a direct flight that saves an hour and a half).
  - it's worth $0.50 for every minute saved waiting in the airport.
- You could also add the cost of an extra day of car rental if everyone returns to the airport at a later time of the day than when they first rented the car.

Computing the Cost (Part 1)

İSTANBUL ŞEHİR ÜNIVERSITY

```
def schedulecost(sol):
    totalprice = 0
    latestarrival = 0
    earliestdep = 24 * 60
    for d in range(len(sol) / 2):
        # Get the inbound and outbound flights
        origin = people[d][1]
        outbound = flights[(origin, destination)][int(sol[2 * d])]
        returnf = flights[(destination, origin)][int(sol[2 * d + 1])]

        # Total price is the price of all outbound and return flights
        totalprice += outbound[2]
        totalprice += returnf[2]

        # Each minute of flight duration costs $1
        totalprice += getminutes(outbound[1]) - getminutes(outbound[0])
        totalprice += getminutes(returnf[1]) - getminutes(returnf[0])

        # Track the latest arrival and earliest departure
        if latestarrival < getminutes(outbound[1]):
            latestarrival = getminutes(outbound[1])
        if earliestdep > getminutes(returnf[0]):
            earliestdep = getminutes(returnf[0])
```

Computing the Cost (Part 2)

İSTANBUL ŞEHİR ÜNİVERSİTY

```
# Every person must wait at the airport until the latest person arrives.
# They also must arrive at the same time and wait for their flights.
totalwait = 0
for d in range(len(sol) / 2):
    origin = people[d][1]
    outbound = flights[(origin, destination)][int(sol[2 * d])]
    returnf = flights[(destination, origin)][int(sol[2 * d + 1])]
    totalwait += latestarrival - getminutes(outbound[1])
    totalwait += getminutes(returnf[0]) - earliestdep

# Does this solution require an extra day of car rental? That'll be $50!
if latestarrival < earliestdep:
    totalprice += 50

return totalprice + totalwait
```

The cost function

İSTANBUL ŞEHİR ÜNİVERSİTY

```
>>> reload(optimization)
>>> optimization.schedulecost(s)
```

Educated Optimization

İSTANBUL ŞEHİR ÜNİVERSİTY

- **Random Searching**
- Hill Climbing
- Genetic Algorithms
- Simulated Annealing

Random Search

İSTANBUL ŞEHİR ÜNİVERSİTY

```
def randomoptimize(domain, costf):
    best = 999999999
    bestr = None
    for i in range(1000):
        # Create a random solution
        r = [random.randint(domain[i][0], domain[i][1])
             for i in range(len(domain))]
        # Get the cost
        cost = costf(r)

        # Compare it to the best one so far
        if cost < best:
            best = cost
            bestr = r

    return bestr
```

Random Searching

İSTANBUL ŞEHİR ÜNİVERSİTY

- This function randomly generates 1,000 guesses and computes the schedule cost for each.
- It keeps track of the best guess (the one with the lowest cost) and returns it.

Random Searching

İSTANBUL ŞEHİR ÜNİVERSİTY

```
>>> reload(optimization)

>>> domain = [(0,9)]*(len(optimization.people)*2)

>>> s =optimization.randomoptimize(domain,optimization.schedulecost)

>>> optimization.schedulecost(s)

>>> optimization.printschedule(s)
```