Problem Set

Trinity University ACM
High School Programming Competition

April 13th, 2013

Problem 0 - Meet Pete

In the busy bustling business world of today, people don't have time to stop and smell the roses. Or determine the best path to take when making a sales trip. Meet Traveling Salesperson Pete (TSP), a member of ACME's sales team. Every week TSP must plan the most efficient route between all the cities in his territory he is visiting to make sales. TSP knows the distance between every city in his territory by heart. Given a list of locations that TSP must visit and the distances between each of those locations, what is the shortest distance that TSP will have to drive starting in his home city, visiting each city once and only once, and returning home given his territory?

Input: The first line of the input will contain a single integer, $0 < N \leq 10$ indicating the number of input sets. Each input set will contain a single integer, $0 < K \leq 12$ indicating the number of cities TSP must visit in his territory that week. There will then be K lines, each containing the name of one city that TSP must visit followed by the distance from that city to all cities following it in the list. The first city will always be his hometown, and each city name will only be a single word.

Output: The output for this problem will be one line per input sets with the minimum number of miles that must be traveled for the TSP to visit every city on the list and return to his starting location without visiting any city more than once.

Sample Input:
```
2
3
Toontown 3 5
Westford 4
Georgetown
5
Toontown 4 4 4 4
Paton 3 5 5
Westend 10 5
Anterson 8
Paulson
```

Sample Output:
```
12 Miles
21 Miles
```

Problem 1 - Out of Sorts

Traveling Salesperson Pete (TSP) always makes sure to give his customers receipts for their purchases. Unfortunately, since he's always in a hurry to get to his next destination, TSP's copies of the receipts are all out of order. He has until tomorrow to get the receipts in order before he turns them in to his boss, Manager Mike. Mike wants to know what items TSP made the most money on. Write a catalog of the receipts for Mike to look at so he can see the total amount of money made from the sale of each kind of item.

Input: The first line of input will contain a single integer, $0 < N \leq 10$, indicating the total number of receipts. Each receipt will contain a single integer, $0 < K \leq 10$, indicating the number of items. There will then be K lines, each containing an item. Item lines will have name, price and quantity sold, in that order and separated by spaces. Each item name will have no white space.

Output: A list using the same datatype that has the items in order (largest to smallest) based on the total revenue (before factoring in cost). The output of this problem should consist of a line for each item with the name, price ($), number of items sold, total revenue earned for that item. Item totals will be unique. There should be a final line containing the total revenue for all items.

Sample Input:
```
2
3
Leg_Lamp 7.40 5
Daisy_Red_Ryder_Air_Rifle 64.14 2
Bunny_Suit_Pajamas 120.00 1
2
Deluxe_Fruitcake 34.19 1
Red_and_Green_Foil_Gift_Wrap 19.99 4
```

Sample Output:
```
Daisy_Red_Ryder_Air_Rifle 64.14 2 Revenue: 128.28
Bunny_Suit_Pajamas 120.00 1 Revenue: 120.0
Leg_Lamp 7.40 5 Revenue: 37.0
Total Revenue: 285.28
Red_and_Green_Foil_Gift_Wrap 19.99 4 Revenue: 79.96
Deluxe_Fruitcake 34.19 1 Revenue: 34.19
Total Revenue: 114.15
```

Problem 2 - It's a Trap!

Traveling Salesperson Pete (TSP) is a serious businessman, with a serious agenda. He doesn't have time to deal with such petty matters such as speed limits and other traffic laws as his trips to and from places of business often need to be very fast. In order to avoid speeding tickets, in his tiny town of Toontown, USA, he has talked to an old buddy of his with a police radio scanner, who has, for an inconsequential fee, agreed to let TSP in on where the current speed traps are, so that he can avoid them. Using this information TSP wants to know if he can finish his route in time to make it home to watch his favorite television show without getting a traffic ticket. TSP's car can travel up to 120 mph, and accelerates instantaneously to his required speed (being a salesman for ACME has it's perks).

Input: The first line of the input will contain a single integer, $0 < N \leq 20$ indicating the number of input sets. Each input set will first contain a single integer, $0 < K \leq 12$ indicating the number of different roads TSP must visit on his route. The next line will be another integer with the time in minutes that TSP needs to reach the end location. Then, for the next K lines, there will be a "C" if there is a cop on this road, or a "N" if there is not. On the same line will also be the distance TSP has to travel and the name of the road which will contain no spaces and the integer speed limit on the road which will be greater than 0.

Output: for each input set you must return Yes if TSP is able to complete the route in time, or No if he cannot.

Sample Input:

```
2
4
60
C 20.0 Cedar 60
N 10.5 Straightshot 30
N 20.0 Powderkeg 45
N 10.8 Charleston 30
1
20
C 30.0 Charleston 30
```

Sample Output:

```
Yes
No
```

Problem 3 - Grab n' Go

On his way out for a sale, Traveling Salesperson Pete (TSP) spots the car of his biggest competition Bandit Keith on the side of the road with a smoking radiator. After pulling over TSP sees that the car is abandoned, obviously Bandit Keith has been forced to walk to town to get help. Seeing an opportunity to expand his profits, TSP quickly looks through Bandit Keith's trunk for items of value. Knowing that he has a limited amount of time until Keith returns and limited space in his trunk, TSP needs to figure out how much of a profit he can make given the items in Bandit Keith's car.

Input: The first line of the input will contain the number of inputs, $0 < N \leq 10$. The first line of each input contains three integers, S, T, and I. S indicates the amount of free space in TSP's trunk. T indicates the amount of time, in minutes, that Pete has until Bandit Keith returns. $0 < I \leq 15$ indicates the number of items in Bandit Keith's trunk. Each item will be a new line consisting of the name of the item, the value of the item, the space the item takes up and the time, in minutes, it will take to move the item.

Output: For each input set the output should be the maximum value that TSP can get from the items he moves from Bandit Keith's car given the time and space restraints.

Sample Input:
```
2
5 10 4
Kerrigan_Dolls 2 2 1
PowerPuff_Vitamin_Pills 50 1 1
Yogi_Ties 10 1 2
Bunny_Slipper_Socks 3 2 3
30 30 2
Chinese_Checkers 2 50 14
Tungsten_Steel_Pliers 30 1 3
```

Sample Output:
```
63
30
```

Problem 4 - Crazy Train

Traveling Salesperson Pete must go through a neighborhood with a train track running across its length. Pete needs to get to the next city on his schedule as soon as possible; he's running late due to an inefficient program that wrote down his itinerary. Luckily, Pete is an amazing judge of speeds and can tell how fast a train is moving just by looking at it. Given that knowledge and the distances that both he and the train have to travel, he needs to figure out if he will make it across the tracks or not.

Input: Input will first consists of a number 0<N<100, this is the number of input sets that you will be provided. Each input set will consists of 4 lines, each with a single, positive integer value. The first line will contain the speed of Pete's car in mph, the second line will contain the distance from Pete's car to the train tracks in miles, the third line will contain the speed of the train in mph, and the fourth line will contain the distance from the head of the train to the intersection of the tracks and the road in miles.

Output: Output should be Yes if Pete will reach the tracks before the train and No if the train will reach the road before Pete. The train wins ties.

Sample Input:
```
2
50
100
60
300
30
100
50
50
```

Sample Output:
```
Yes
No
```

Problem 5 - Are we there yet?

Traveling Sales Pete used an inefficient path to sell his goods in various cities and ended up driving for a long time. He realized today that he needed to drive less and spend more family time. He said out loud, "I can finally go home, nothing is going to delay me from seeing my family." Then the heavy rain started.

The rain is making it difficult for Pete to read the exit signs. He needs to pick the right exit to get home to his family as soon as possible to set things right. The tears and the rain make it difficult to see every character in the road signs. As he drives near each exit, he manages to only catch a few characters of the signs.

Given the characters Pete sees of each sign of the exits he takes and his map information, determine if it is possible that Pete can go home safely and see his family.

Input: The first line of the input will contain a single number, $0 < N < 20$ indicating the number of input sets. The The first line of each input set will have two city names representing where Pete is and where home is. That will be followed by a line with a number, $0 < R \leq 40$, for the number of roads on the map. The following R lines will each have two words for two cities connected by a highway. Note that these roads are all two way. These will be followed by another number, $0 < S \leq 8$, telling how many signs Pete saw on his trip. The input for each set ends with S lines giving what he saw. Characters he couldn't make out are shown as hyphens. Signs that are seen will always be possible road combinations.

Output: The output, for each input set, should be "Maybe home." if it is possible that the set of exits Pete took got him home and "Definitely lost." otherwise. Note that Pete sees a sign for every exit he takes. So he will always see a sign in going from one city to another.

Sample Input:

```
2
Austin Houston
4
Austin Dallas
Dallas Waco
Houston Temple
Waco Temple
2
D-ll-s
W-c-
Austin Houston
2
Austin Houston
Austin Halifax
1
H------
```

Sample Output:

```
Definitely lost.
Maybe home.
```

Problem 6 - Forget the rules, I've got money!

Traveling Salesman Pete ends up in a confrontation with his rival, Bandit Keith. Like all sane businessmen, they decide to settle their dispute in the only way they know how. A children's card game. Each player is dealt 5 cards. Each card is either a monster card, with an attack strength, or a trap card. The winner is the player with the highest total monster strength in their hand, however each trap card will cancel the strength of the opponents strongest remaining card. Pete, not one to leave things to chance, has a card up his sleeve when he plays these games. He has the option of substituting one of his cards for his hidden card if it will help him not lose.

Input: The first number of the input will be the number of sets. The next 5 lines will contain the name of the card, no space between words, and a number, the attack strength, if it's a monster card or the word "Trap" if it's a trap card. The next card will be Pete's hidden card. The last 5 cards in each input set will be the 5 cards in Bandit Keith's hand, following the same format as the other cards.

Output: The output should be "Win" if Pete will win with his original hand, "Lose" if Bandit Keith will win with his original hand even if Pete cheats, "Tie" if Pete will tie against Keith without cheating and "Cheat" if Pete will not lose - win or tie - if he switches one of his cards with the card in his sleeves.

Sample Input:

```
2
The_Dark_Magician 500
Toy_Soldier 100
Prison_of_Bone  Trap
Heaven_Cannon 300
Magician’s_Apprentice 200
Sandpit Trap
Goblin_Archer 400
Goblin_Techies 600
Golden_Warrior 100
Panda_Monk 400
Sarlak Trap
Doctor_Hooligan 300
Mister_Freeze 300
Dark_Bell_Curve 100
Mythical_Hair 100
Rambunctious_Rhino 700
Dull_Secret_Agent 100
Swiper_Fox 100
Knapsack_Of_Holding Trap
Jumping_Magic_Beans Trap
Lukewarm_Water Trap
Ornery_Owl 200
```

Sample Output:

```
Cheat
Lose
```

Problem 7 - What happens in Vegas...

Traveling Salesperson Pete (TSP) is happily married man. Well usually. Right now his wife is upset that he spends so much time on the road and is demanding that he take her on vacation to Las Vegas. Since he values his life, TSP is doing whatever he can to make this happen. Given a list of TSP's inventory, the monthly orders that he gets, his monthly expenses, and the cost of the vacation you need to determine the number of months that it will take TSP to save up enough money to take the vacation. It is also possible that he will be unable to save enough money or it will take too long and his wife's patience will expire.

Input: The first line will contain a single number, $0 < N < 20$ indicating the number of input sets. The first line of each input set contains an integer, $0 < M \leq 10$ indicating the number of items in TSP's catalog. The next M lines will be the items that TSP sells in alphabetical order by the name of each item, and the cost to TSP to order the item. There will then be a number of sales, $0 < K \leq 10$, of items each month. This will be followed by K lines of sales. Each item sale will consist of the name of the item sold, the number of items sold, and the value of each item individually. This list will not be in a specific order and can contain multiple orders for the same item or no orders for some items. The next line will be the expenses that TSP needs to pay each month. The final line will be the cost of the trip.

Output: The number of months it will take to earn the required amount. If TSP is unable to take a vacation, or if it will take longer than 12 months return "R.I.P." instead.

Sample Input:

```
2
2
Hearing Aid $50
Stuffed Penguin $30
2
Hearing Aid 5 $200
Hearing Aid 3 $150
$400
$5000
1
Jumbo Jet $30
1
Jumbo Jet 1 $15
$200
$1000
```

Sample Output:

```
8
R.I.P.
```

Problem 8 - PizzaDiction = Pete's Addiction

Pete is addicted to Pizza. He is so addicted that he can only travel for so long without eating a slice. His wife has begged him to go to a professional but he refuses. He leaves his house in a puff and doesn't get to plan his travel route as he sells his goods like he usually does.

Given his itinerary, along with the distance between pizza stores and the number of pizzas his car can hold, will TSP be able to complete his trip or will he give up before finishing due to a lack of pizza? Each pizza box contains 8 slices of pizza and TSP needs a slice of pizza every so often.

Input: The first line will contain a single number, $0 < N < 20$ indicating the number of input sets. The first line of each input set will have four integer values. These represent the number of pizza boxes Pete's car can hold, the number of miles TSP can travel for each slice of pizza he eats, the number of pizza stores on his route ($0 < S \leq 10$), and the total distance he has to go on his trip. Then the S lines after that will have the name of each pizza store followed by distance Pete needs to travel to get to it from the previous location. The first pizza store will always have a distance of 0. Names of pizza stores will contain no spaces.

Output: Each line of the output should say "BLISS" or "AGONY" depending on if Pete manages to obtain enough pizza for each input set. "BLISS" if there is enough pizza so that he can finish his itinerary or "AGONY" if he stops trying because he runs out of pizza.

Sample Input:

```
2
4 10 2 520
Pizza_Palace 0
Papas_Pizza 240
3 2 4 110
Lisa_Pizza 0
Little_Ceaser 20
Testing_Pizza_Palace 50
Johns_Dine_And_Dash 30
```

Sample Output:

```
BLISS
AGONY
```

Problem 9 - Pete's Peddlers

Pete's empire is expanding and he is finding it hard to personally make deliveries to all of the different cities. The items he sells all arrive in his home city and it would really help to just get things closer to where they need to end up going. Pete is far too cheap to hire real delivery drivers to do this. Instead he has come up with the idea of paying High School students sub-minimum wage to ride their bikes and transport items for him. Even with this low cost approach, Pete still trying to further optimize his costs.

Pete pays the students per mile that they ride between cities. He doesn't really care how far the items go when he isn't driving them, he just needs to be able to get items from the source city to every other city. So he wants you to find the roads to include in order to spans all the cities, but with the minimum total distance. For this problem you need only tell him the length of the sum of the roads in miles.

Input: The first line of the input will contain a single number, $0 < N < 20$ indicating the number of input sets. Each input set will contain a single number, $0 < K \leq 20$ indicating the number of cities TSP needs to connect. There will then be K lines, each containing the name of one city followed by the names and distances from that city to all nearby cities that can be ridden to on a bike. The distances will be integer values and will be symmetric. The first city will always be his hometown, and each city name will only be a single word.

Output: The output for this problem will be one line per input sets with the minimum number of miles of road that he needs to pay the students to cover.

Sample Input:

```
2
3
Toontown Westford 3 Georgetown 5
Westford Toontown 3 Georgetown 4
Georgetown Toontown 5 Westford 4
5
Toontown Paton 4 Westend 4
Paton Toontown 4 Anterson 3 Paulson 5
Westend Toontown 4 Paulson 10 Anterson 5
Anterson Paulson 8 Paton 3 Westend 5
Paulson Paton 5 Westend 10 Anterson 8
```

Sample Output:

```
7
16
```

Problem 10 - Distribution Difficulties

Pete has been expanding his courier network and he needs to know if the current network can handle the orders he is getting. All of his orders come in at one city and go out to the others. He is paying kids to bike things between cities and he knows how much weight can go between each pair of cities in his network each day. He doesn't care exactly how long it takes for things to get from the delivery spot to their destination, as long as the courier network can keep up so that in a steady state there aren't items piling up in any given city.

You are given the roads he has couriers working on and how many pounds of inventory can be carried down each road per day. These have a direction as Pete has arranged for couriers to only carry inventory one way and just ride back not carrying anything. You also have how many pounds of items are to be sold in each city each day. Assume the number of pounds received in Pete's home town is equal to the total orders. Your program needs to tell Pete if the courier network can handle the orders.

Input: The first line of the input will contain a single number, 0 < N < 20 indicating the number of input sets. Each input set will start with an integer, 0 < K ≤ 20, indicating the number of cities in the courier network. There will then be K lines, each containing the name of one city followed by the number of pounds of items that city need each day, 0 < P < 1000, and connection information. The connection information is in the form of city names and courier capacities from that city to all nearby cities with connecting roads. The first city will always be his hometown and it will have zero pounds of orders to consider. Each city name will only be a single word. There will never be couriers running in opposite directions between two cities.

Output: Each input set will have a single line of output. If the network of couriers can handle the flow of products print "SUCCESSFUL". Otherwise you should print "TOO HEAVY".

Sample Input:

```
2
3
Toontown 0 Westford 3 Georgetown 5
Westford 3 Georgetown 4
Georgetown 2
5
Toontown 0 Paton 4 Westend 4
Paton 6 Anterson 3
Westend 2 Paulson 10 Anterson 5
Anterson 4 Paulson 8
Paulson 3 Paton 5
```

Sample Output:

```
SUCCESSFUL
TOO HEAVY
```