

Problem Set 2

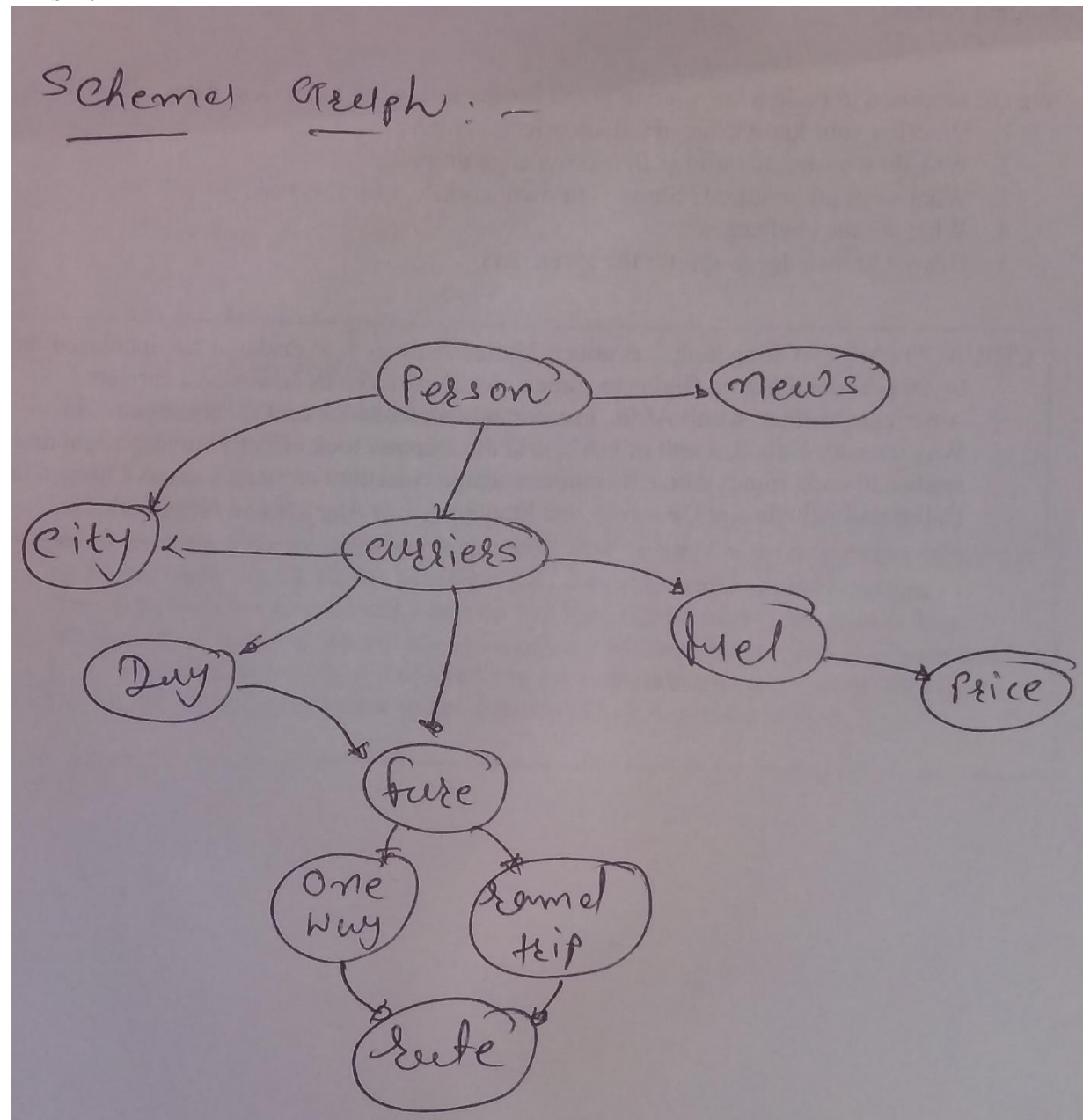
Knowledge Discovery and Management

Class Id: 21

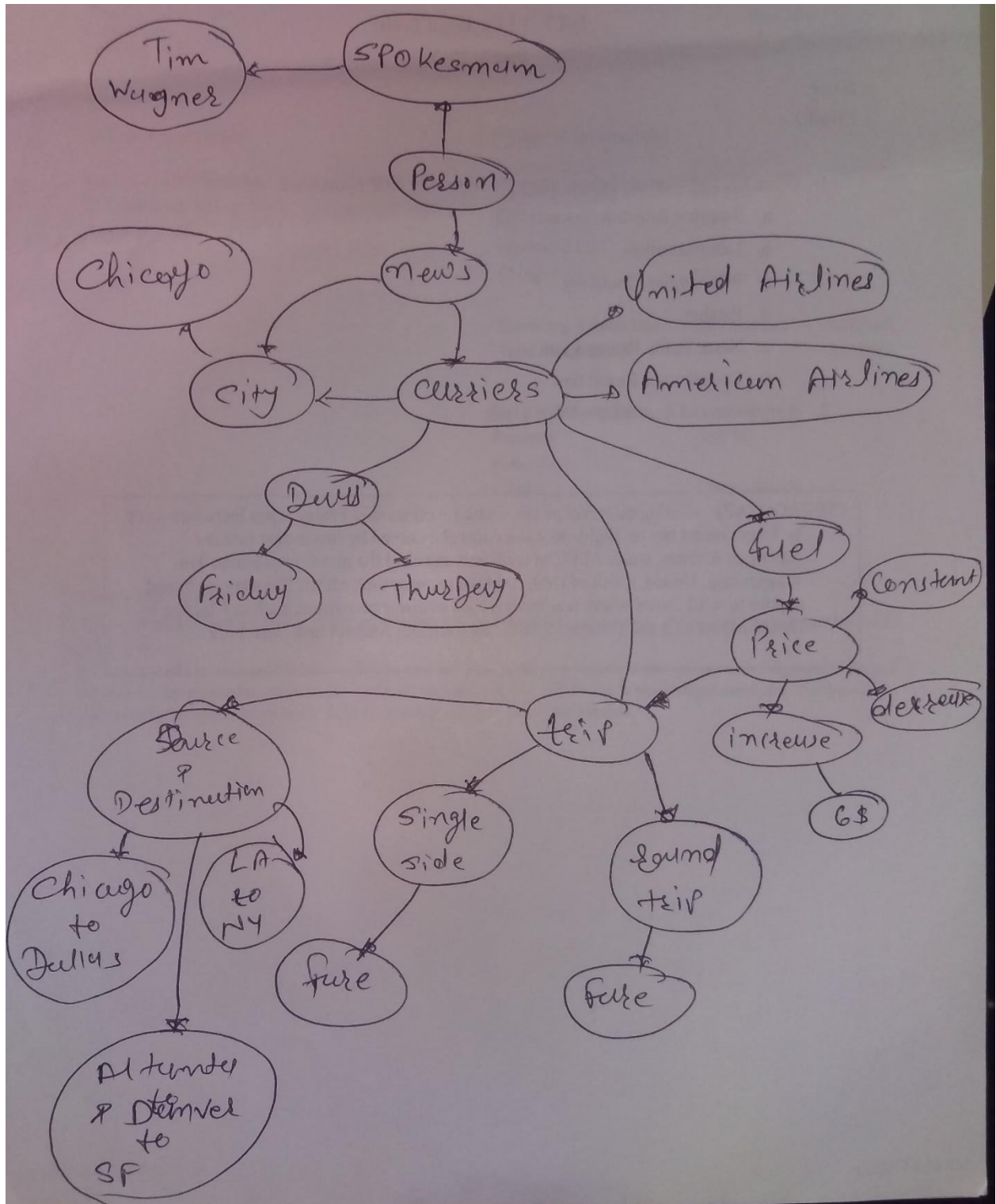
Name: Shalin Patel

PS-2A

Ans 1:



Datagraph



Question and Ans Template:

1. What:

What +verb+ noun -> Ans: Noun + verb + ans

Que: What was increment in rate?

Ans: Increment in rate was \$6.

2. When:

When + noun+ verb? -> From + ans

Que: When increase in rate will apply?

Ans: From Thursday midnight.

3. Where:

Where + verb + noun -> Noun+ from +Ans

Que: Where is the news from?

Ans : The news is from Chicago.

4. Who:

Who + verb + noun -> Ans+ verb + noun

Que: Who was the spokesman?

Ans: Tim Wanger was the spokeperson.

5. How:

How + verb + noun + adjective -> By + verb + ans

Que: How rise in price will compensate?

Ans: By increasing in fare of 6\$

6. How many:

How many + noun + verb -> Ans + verb

Que: How many airlines were mentioned?

Ans: Two airlines were mentioned.

7. Yes/No:

Question -> If words matches, Yes otherwise No

Que: Rate change affected immediately?

Ans: No

PS - 2B

Ans 1:

i/p

A → B c D
B → A c D E
C → A B D E
D → A B c E
E → B c D

Spitting

A B c D
B A c D E
C A B D E
D A B c E
E B c D

mapping

(A, B), C
(A, B), D
(B, A), C
(B, A), D
(B, A), E
(C, A), B

(C, A), D
(C, A), E
(D, A), D
(D, A), C
(D, A), E
(E, B), C

Shuffling & Reducing

(A, B), (C), (D)
(A, B), (D), (E)
(B, A), (C), (D)
(B, A), (D), (E)
(C, A), B

(C, A), (D), (E, B), (D)
(C, A), (E)
(D, A), B
(D, A), (C)
(D, A), (F)
(E, B), (C)

Result

(A, B) → (C, D)
(A, C) → (B, D)
(A, D) → (B, C)

(B, C) → (A, D, E)
(B, D) → (A, C, E)
(B, E) → (C, D)
(C, D) → (A, B, E)
(C, E) → (B, D)
(D, E) → (B, C)

Ans 2: Algorithm

```
method Map ( list of doc)
for each term  $t \in$  list
  for  $i$  to  $t.length - 1$ 
    Emit( $t(0)$ ,  $t(1)$ ,  $t(i)$ )

method Reduce ( list )
for each terms  $t \in$  list
  String String.Concat( $list(0)$ ,  $list(1)$ ,
                         $list(2)$ )
```


Ans 3: Map Reduce Function

```
def myMap (lst: List): List = {  
    var var mappedList:  
    var i = 0;  
    for (l < lst.length) {  
        for (i < 2 to lst(l).length) {  
            mappedList :+ List(lst(l)(0),  
                                lst(l)(1), lst(l)(i))  
        }  
    }  
}
```



```
val textFile = spark.textFile("...")  
val rddFriends = textFile.flatMap(  
    line => line.split(" ") /  
    map (lst => def myMap(lst)).  
    reduceByKey (lst(0) -> "+" + lst(1)  
                 -> "+" + lst(2))
```