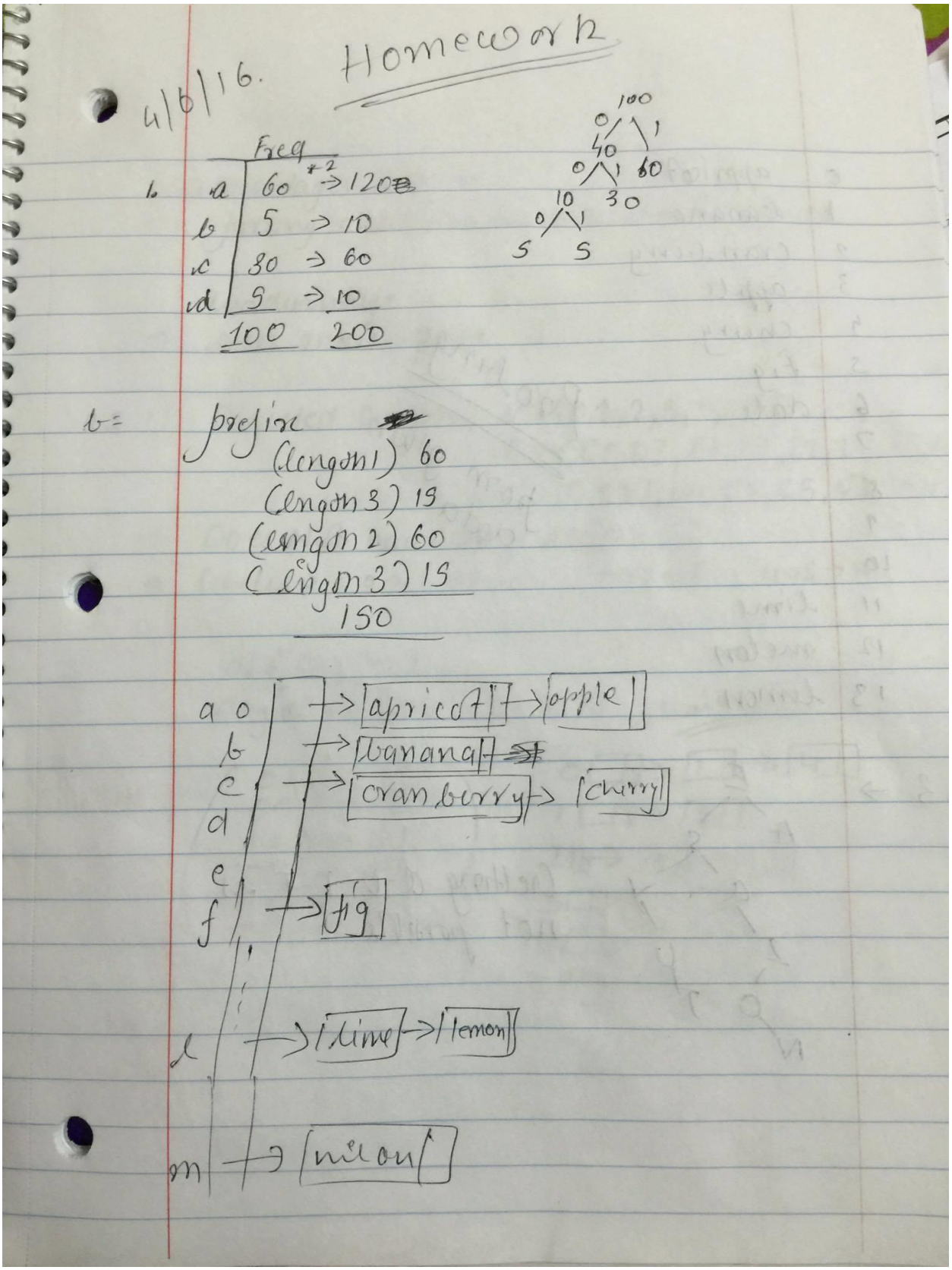
CS323- LECTURE NOTES
4/7/2016 - aishwarya



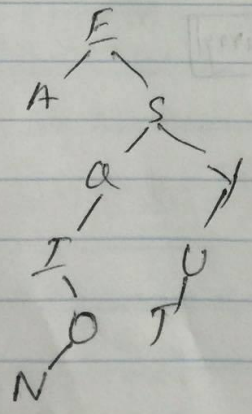
CS323- LECTURE NOTES
4/7/2016 - aishwarya

0	apricot
1	banana
2	cranberry
3	apple
4	cherry
5	Fig
6	date
7	
8	
9	
10	
11	lime
12	melon
13	lemon

Probing

from given data

Q3 →



Getting Q-U-I-T is not possible.

CS323- LECTURE NOTES
4/7/2016 - aishwarya

advantages of trie
- finding characters are easier and is quick.

disadvantage
→ takes more space.

a.5 1) Directed Graph $V = \{0, 1, 2, 3, 4, 5\}$

$E = \{[0,1], [1,2], [2,1], [2,0], [3,2], [4,5], [5,4], [0,4]\}$

→ Does Not have any loops

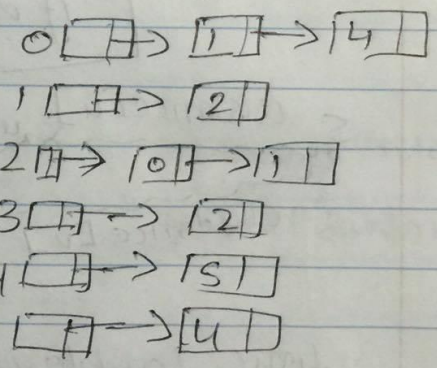
→ Cyclus → $0 \to 1 \to 2 \to 0$ $1 \to 2 \to 1$ $4 \to 5 \to 4$

$deg^-(V_2) = 2$

$deg^+(V_2) = 2$

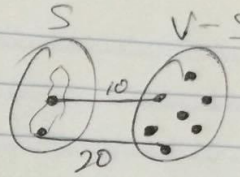
$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

chaining



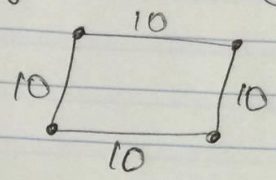
CS323- LECTURE NOTES
4/7/2016 - aishwarya

Prim's Algorithm
Kruskal's Algorithm



its a graph so vertices are connected.

min cost



do proof by contradiction

min cost → cost of edge
closest

```
for i=1 to n
    min cost[i] = C[s,i]
    closest[i] = s

for i = 2 to n          (remaining vertices in V-S)
j = Find closest minimum of min cost
    for vertices still in V-S.
    for each neighbours K of j in V-S
        if min cost[K] > C[j,K]
            min cost[K] = C[j,K]
            closest[K] = j
```

time complexity $O(|V|^2 + |V|^2)$ → improve (heap) $(O((|V|^2 + |E|))$

finding minimum ↓ loop ↓ $\rightarrow |V| \log |V| + |E| \log |V|$

CS323- LECTURE NOTES
4/7/2016 - aishwarya

Class work

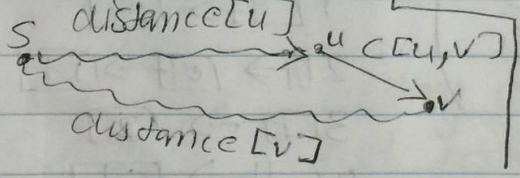
Dijkstras SSSP Algorithm

distance[w] = shortest path known between source (s) and w

pred[w] = the vertex immediately before w on the path from s to w

Init {
for vertices $w \in V$
distance[w] := ∞
pred[w] = s
end for
}

while q (priority min-heap queue) is not empty
u = getMin(q)
for each neighbour v of u
if distance[u] + c[u,v] < distance[v]
distance[v] = distance[u] + c[u,v]
pred[v] = u
update q - priority of v



time complexity → $O(|E| \log |V| + |V|)$

delete mean V times.
$O(|E| + |V| \log |V|)$

Amortized Analysis.

CS323- LECTURE NOTES
4/7/2016 - aishwarya

$50 ⟶ $100
10 years.

APSP = call pairs shortest path
(Single source shortest Path can be applied)

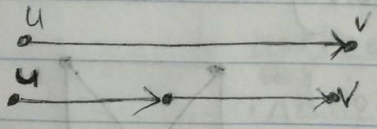
for $v \in V$
runDijkstra's Algorithm with source v

Dynamic Programming

$f(0) = 0 \quad f(1) = 1$
$f(n) = f(n-1) + f(n-2)$

```
f[0] = 0
f[1] = 1
for i = 2 to 100
    f[i] = f[i-1] + f[i-2]
```

hybrid → memoization



introduce intermediate vertex
optimal substructure

wrong way →
```
for i = 1 to n
    for j = 1 to n
        for k = 1 to n
```

CS323- LECTURE NOTES
4/7/2016 - aishwarya