CSCE 310H - Fall 2021

Computer Science III - honors

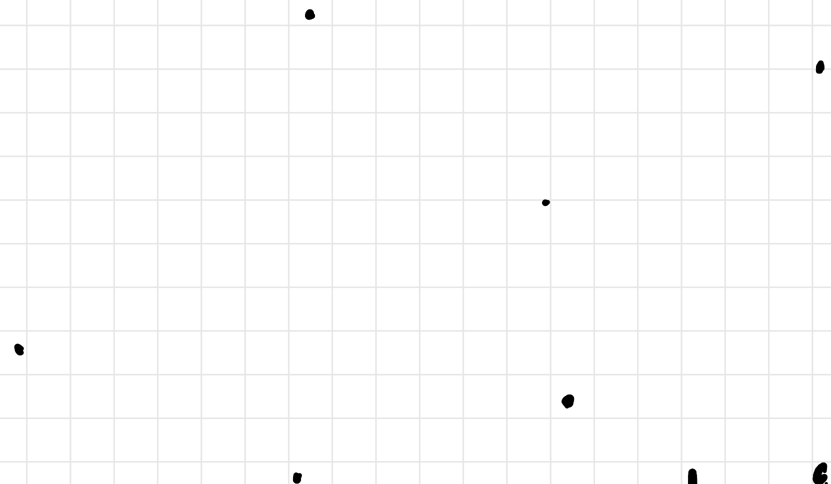
Dr. Chris Bourke

(x_1, y_1)

d

(x_2, y_2)

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$



Pairs = Combinations of size 2

given a set of n distinct elements, how many combinations (order does not matter) of k elements are there?

n "choose" k

$$\binom{n}{k} = C(n,k) = C^n_k = {}_nC_k$$

$$\binom{n}{k} = \frac{n!}{(n-k)!\,k!}$$

ex:

$$\binom{n}{2} = \text{number of pairs}$$

$$= \frac{n!}{(n-2)!\,2!} = \frac{n(n-1)}{2} \in O(n^2)$$

$$= \frac{n^2}{\cancel{2}} - \frac{\cancel{n}^2}{2}$$

Input: A set of points $A = \{(x_1, y_1), \dots (x_n, y_n)\}$

Output: 2 closest points in A

```
1  MinDist ← ∞
2  for each pair of points (x_a, y_a), (x_b, y_b) in A
3      d ← sqrt((x_a - x_b)^2 + (y_a - y_b)^2)
4      if d < MinDist
5          MinDist ← d
6          (p_a, p_b) ← (p_1, p_2)
7  Output (p_a, p_b)
```

Line 2: $\overset{P_1}{(x_a, y_a)}, \overset{P_2}{(x_b, y_b)}$

Line 3: $d \leftarrow \boxed{\sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}}$

Line 4: if $d \boxed{<} \text{MinDist}$

1) Input: A

2) Input size: n

3) Element Op: comp, line 4

4) $\binom{n}{2}$

5) $\Theta(n^2)$

~~for x in A:~~

~~for y in A:~~

O loose

$n \in \Theta(n)$

$n \in O(n^2)$

$n \in O(2^n)$

for $i = 0 \ldots n-1$

for $j = i+1 \ldots n$

...

1

$$\sum_{i=0}^{n-1} \sum_{j=i+1}^{n} 1 = \sum_{i=0}^{n-1} (n-i)$$

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$

Gauss's Formula

K-combinations

Given a set of n elements, $\{1, 2, 3, \dots n\}$

a k-combination is a subset of size k

↳ unordered

ex) $\{a, b, c, d\}$

2 combos:

a, b   b c   c d

a c   b d

a d

6

$$\binom{4}{2} = \frac{4!}{(4-2)!(2!)} = \frac{4 \cdot 3}{2} = 6$$

3 - combos of a b c d ... n

for i = 1 ... 2 $^{n-2}$

for j = i+1 ... 3 $^{n-1}$

for k = j+1 ... 4 ^{n}

Want: generate evy possible subset...

- how mny subsets are there of a set of size n?

$$\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \binom{n}{3} \cdots \binom{n}{n-1} + \binom{n}{n}$$

$$= 2^n$$

$O(2^n) \rightarrow$ exp.

$S = \{a, b, c \cdots n\}$

$P(S)$ = power set

a, b, c	bit		
$\emptyset$	000	000	0
$\{a\}$	100	001	1
$\{b\}$	010	010	2
$\{c\}$	001	011	3
$\{a\ b\}$	110	100	4
$\{a\ c\}$	101	101	$\vdots$
$\{b\ c\}$	011	110	
$\{a, b, c\}$	111	111	7

0 if x is not in th subset

1 if $x \in$ Subset

$0 \ldots 2^n - 1$

$$\binom{n}{k} = \frac{n!}{(n-k)!\,k!}$$

$$\binom{n}{k} \in O(n^k)$$

$$\binom{n}{2} \in O(n^2)$$

$$\binom{n}{3} \in O(n^3)$$

$$\binom{n}{4} \in O(n^4)$$

$1 \quad \binom{0}{0}$

$2 \quad \binom{1}{0} \qquad \binom{1}{1} \quad 1$

$1 \quad \binom{2}{0} \qquad \binom{2}{1} \quad 2 \qquad \binom{2}{2} \quad 1$

$1 \quad \binom{3}{0} \qquad \binom{3}{1} \quad 3 \qquad \binom{3}{2} \quad 3 \qquad \binom{3}{3} \quad 1$

$1 \quad \binom{4}{0} \qquad \binom{4}{1} \quad 4 \qquad \binom{4}{2} \quad 6 \qquad \binom{4}{3} \quad 4 \qquad \binom{4}{4} \quad 1$

$$\binom{n}{n/2}$$

$$\binom{n}{n}$$

$$\binom{n}{n/2} = \frac{n!}{(n-\frac{n}{2})!\,\frac{n}{2}!} \in O(2^n)$$

$\{2, 5, 6, 9, 10\}$

$n = 10$
$k = 5$

$\downarrow$

2 5 7 9 10

Algorithm 2: Next k-Combination

INPUT : A set of n elements and an k-combination, $a_1 \cdots a_k$.

OUTPUT : The next k-combination.

```
i = k
WHILE a_i = n - k + i DO
    i = i - 1
a_i = a_i + 1
FOR j = (i + 1) ... k DO
    a_j = a_i + j - i
```

2) replace $a_1 \to a_1 + 1$ (i)

$a_2 : a_1 + j - i$ (j)

$2 + 2 - 1 = 3$

$a_3 : a_1 + j - i$ (j)

$2 + 3 - 2 = 4$

$n = 5 \quad \{1\ 2\ 3\ 4\ 5\}$

$k = 3$

Next?

1) locate last a_i such that

$a_i \neq n - k + i$

$a_3 = 5 \stackrel{?}{=} n - k + i$ ✓

$= 5 - 3 + 3 = 5$

current:

a_1	a_2	a_3
$\boxed{1}$	4	5
↓	↓	↓
2	3	4

$a_1 = 1 \stackrel{?}{=} 5 - 3 + \cancel{1}$

$\neq 3$

$a_2 = 4 \stackrel{?}{=} 5 - 3 + i$ ✓

$= 5 - 3 + 2 = 4$

Permutations

A permutation is an arrangement (order matters) of elements

abc   bac   cab   in general:

acb   bca   cba $n!$

$n = 3$ 6

$n = 6$

1 2 3 4 5 6

1 2 3 4 6 5

1 2 3 5 4 6

$\underbrace{\text{perm}} \; \underbrace{\text{"next"}} \; \underbrace{\text{sorty}}$

$n = 6$

$a_i \ a_{i+1}$

$1\ 6\ \boxed{3\ 5}\ 4\ 2$

a'

$1\ 6\ 4\ \underbrace{5\ 3\ 2}_{\text{sort}}$

$1\ 6\ 4\ 2\ 3\ 5$ ✓

1) find last pair that is in order

2) find a': smallest element <u>larger</u> than a_i to the right

   a) swap a_i a'

   b) sort everything to the right of a'

1 6 4 2 3 5

$\underbrace{3 \ \ 5}_{\text{swap}}$

1 6 4 2 5 3

$\underbrace{3}_{\text{sort}}$

$\vdots$

6 5 4 3 2 1

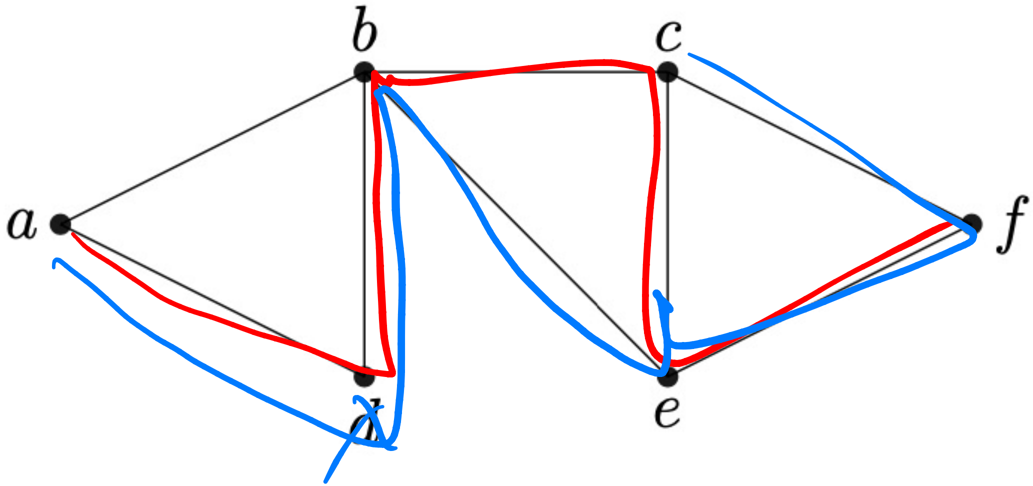
1) right most inorder pair

$n!$ permutations

$$O(n!) = O(n^n)$$

$$0! = 1$$

$, n$



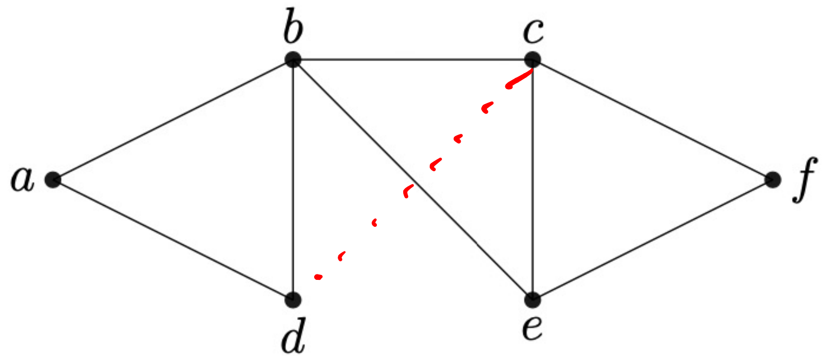
a d b c e f   a b c e f d

a d b e f c

for each permutation π of $\{v_1 \ldots v_n\}$:

    // verify π is a path:
    isValid ← true
    for $i = 1 \ldots n-1$:
        if $(v_i\, v_{i+1}) \notin E$
            isValid ← false
    if isValid
        output Yes
output No

$O(n!)$



$n! = 6! = 720$

$\downarrow$

600

$\downarrow$

480

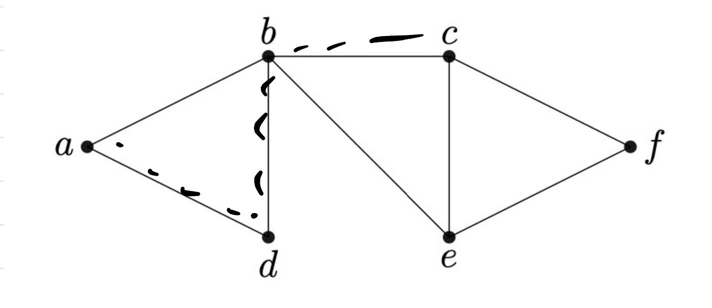
$\hat{x}$

dc , a , b , e , f

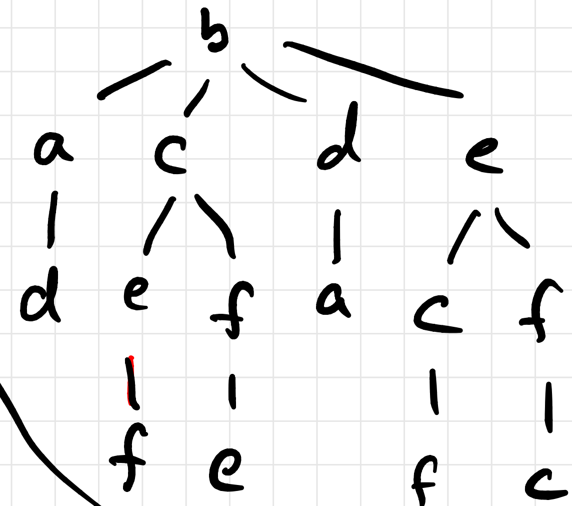
$5! = 120$

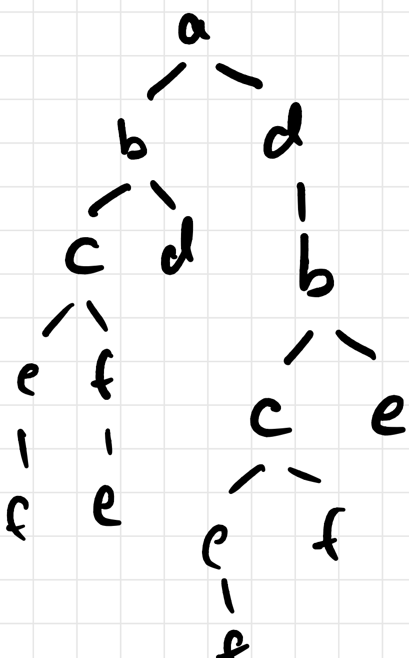
cd

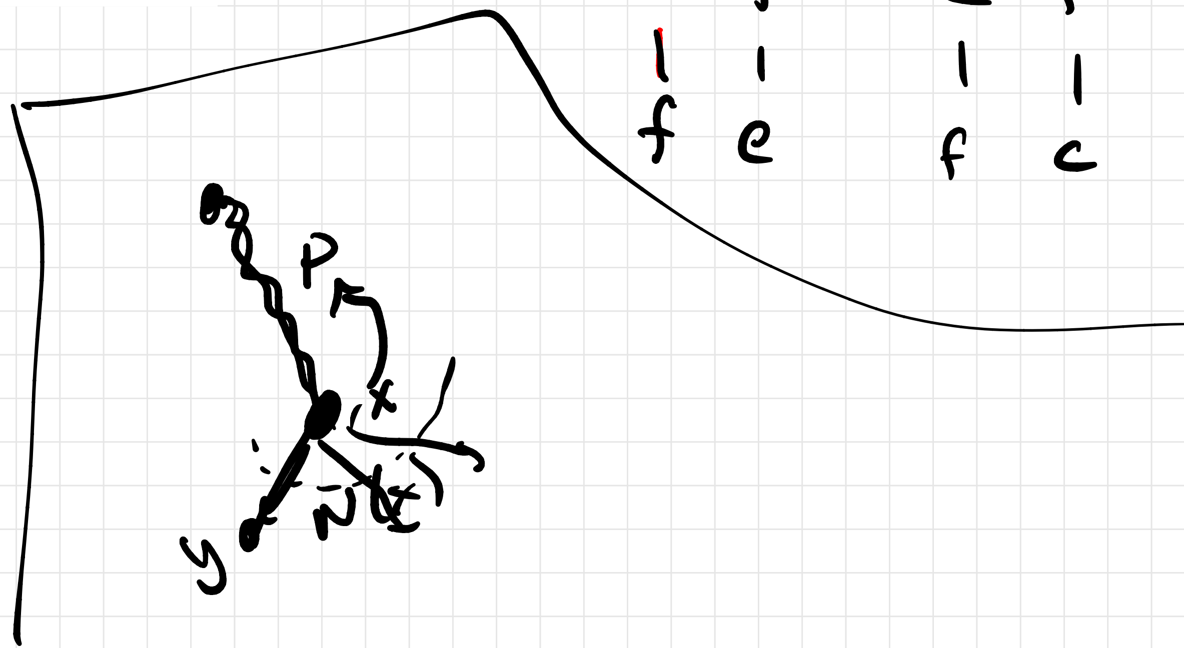
$5!$ 120



a
b
c
d
e
f







P
x
N(x)
y

DFS Hamiltonian Walk – MAIN

Input: A graph $G = (V, E)$

Output: true if G contains a Ham. path

for each $v \in V$:
- path $\leftarrow v$
- if WALK(G, p):
  - output true

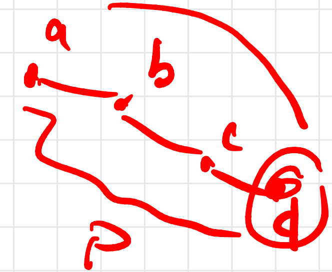
output false

WALK(G, p):

Input: A graph $G = (V, E)$, a path p

Output True if G contains a Ham. path

if $|p| = |V| - 1$
  output true
$x \leftarrow$ last vertex in p
for $y \in N(x)$
  if $y \notin p$
    WALK(G, $p + y$)
Output false



$N(x)$ = neighborhood of x

$|p|$ = length of the path p

$|V|$ cardinality of V

Item	Value	Weight
a_1	15	1
a_2	10	5
a_3	9	3
a_4	5	4

$W = 8$ max capacity

Knapsack

a_1	15	1	(7)
a_3	9	3	(4)
a_4	5	4	(0)
	29		

a_1	15	1
a_2	10	5
	25	6

a_2, a_4 $\to$ 9

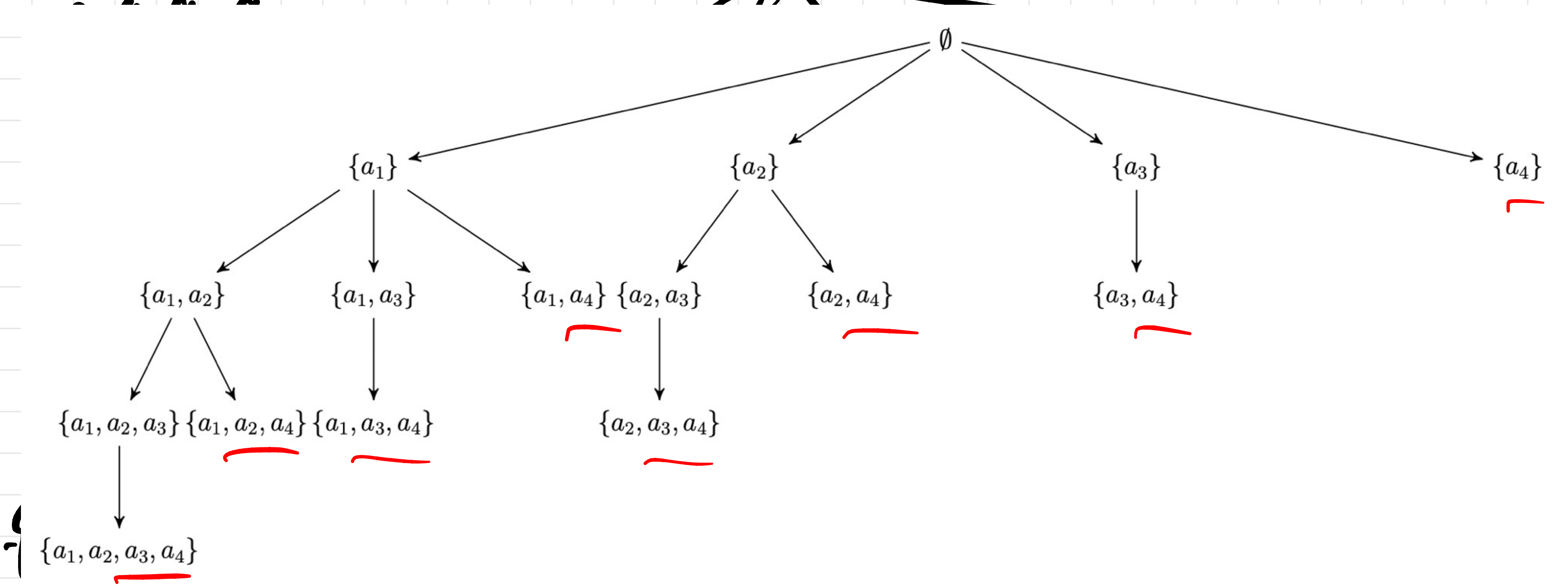
$\boxed{a_2 \; a_4}$ a_1

a_2 a_4 a_3

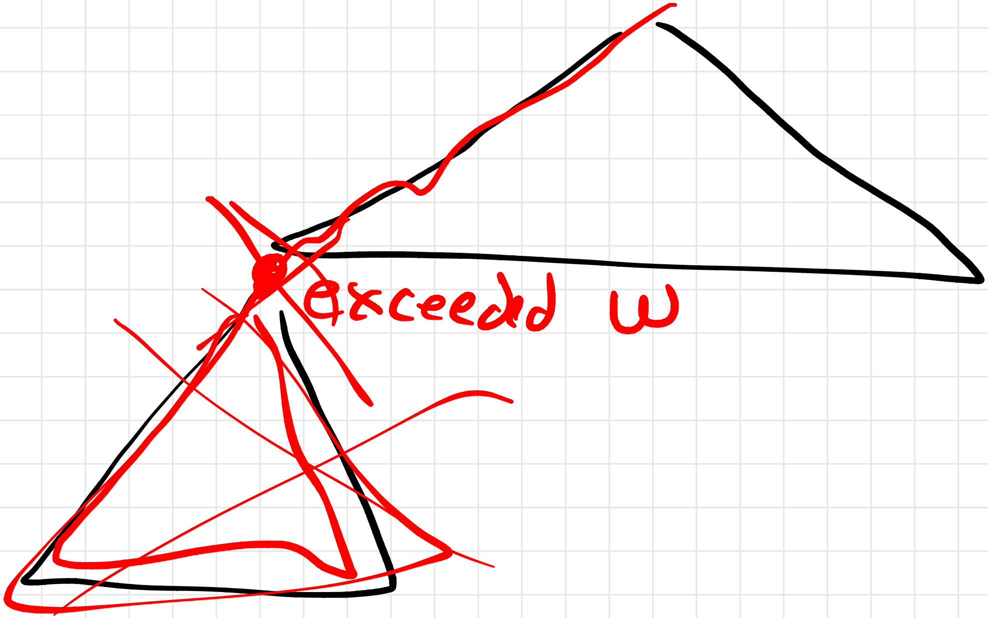
a_2 a_4 a_3 a_1

$n = 4$

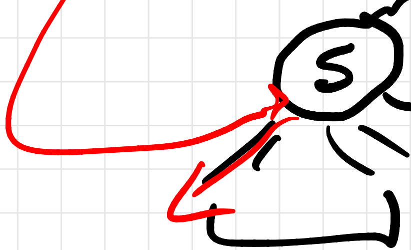
$\emptyset$



$\{a_1, a_2, a_3, a_4\}$


exceed w

$A = \{a_1\ a_2 \cdots a_n\}$



current subset consisting of some elements $a_1 \cdots a_k$

consider adding $a_{k+1}\ a_{k+2}\ a_{k+3} \cdots a_n$

$S \subseteq \{a_1 \dots a_k\}$

add a_{k+1} | add a_{k+2} | ... | add a_n

we need:

S, the subset

A the original

k the index of elements you've considered

KNAPSACK(K, k, S)

Input: An instance of the 0-1 knapsack $K = (A, wt, val, W)$

An index k, a partial solution $S \subseteq A$ not consisting of elements indexed more than k

Output: A feasible solution at least as good as S

if $k = n$
  return S

$S_{best} \leftarrow S$

for $j = k+1 \dots n$
  $S' \leftarrow S \cup \{a_j\}$
  if $wt(S') \leq W$ // feasible
    :
      $T \leftarrow$ KNAPSACK(K, j, S')
      if $val(T) > val(S_{best})$
        $S_{best} \leftarrow T$
      (if)

return S_{best}

sum of values of items in T

sum of wt's of items in S'

a b c α

$\{a\}$ $\{b\}$ $\{c\}$

$\{a\ b\}$ $\{c\}$

$\{a\ c\}$ $\{b\}$

$\{b, c\}$ $\{a\}$

$\{a, b c\}$