DATA MINING 2

Exercises – Sequential Pattern & Clustering

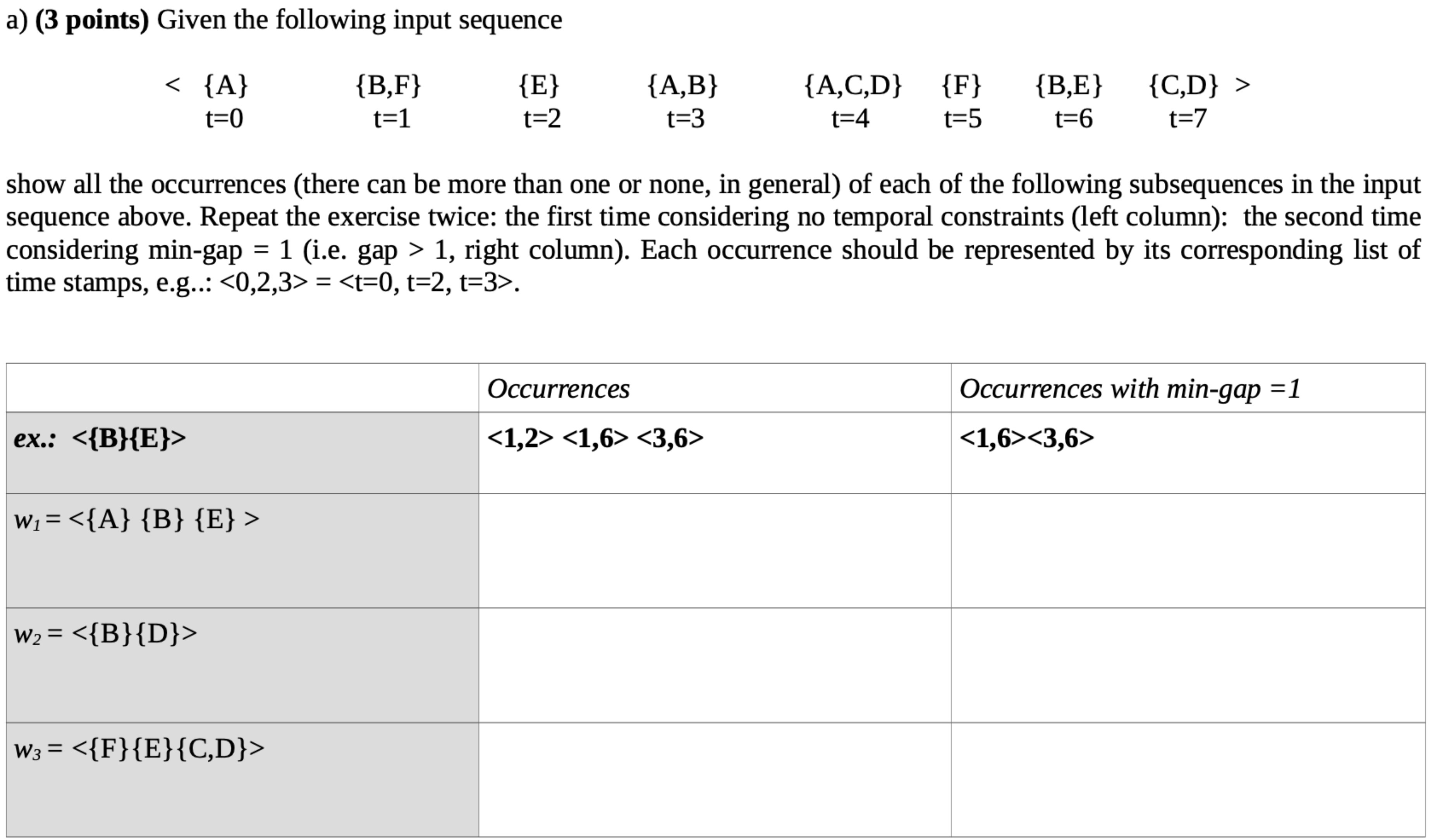
Riccardo Guidotti

a.a. 2019/2020

UNIVERSITÀ DI PISA

Sequential Pattern

Sequential Pattern – Exercise 1



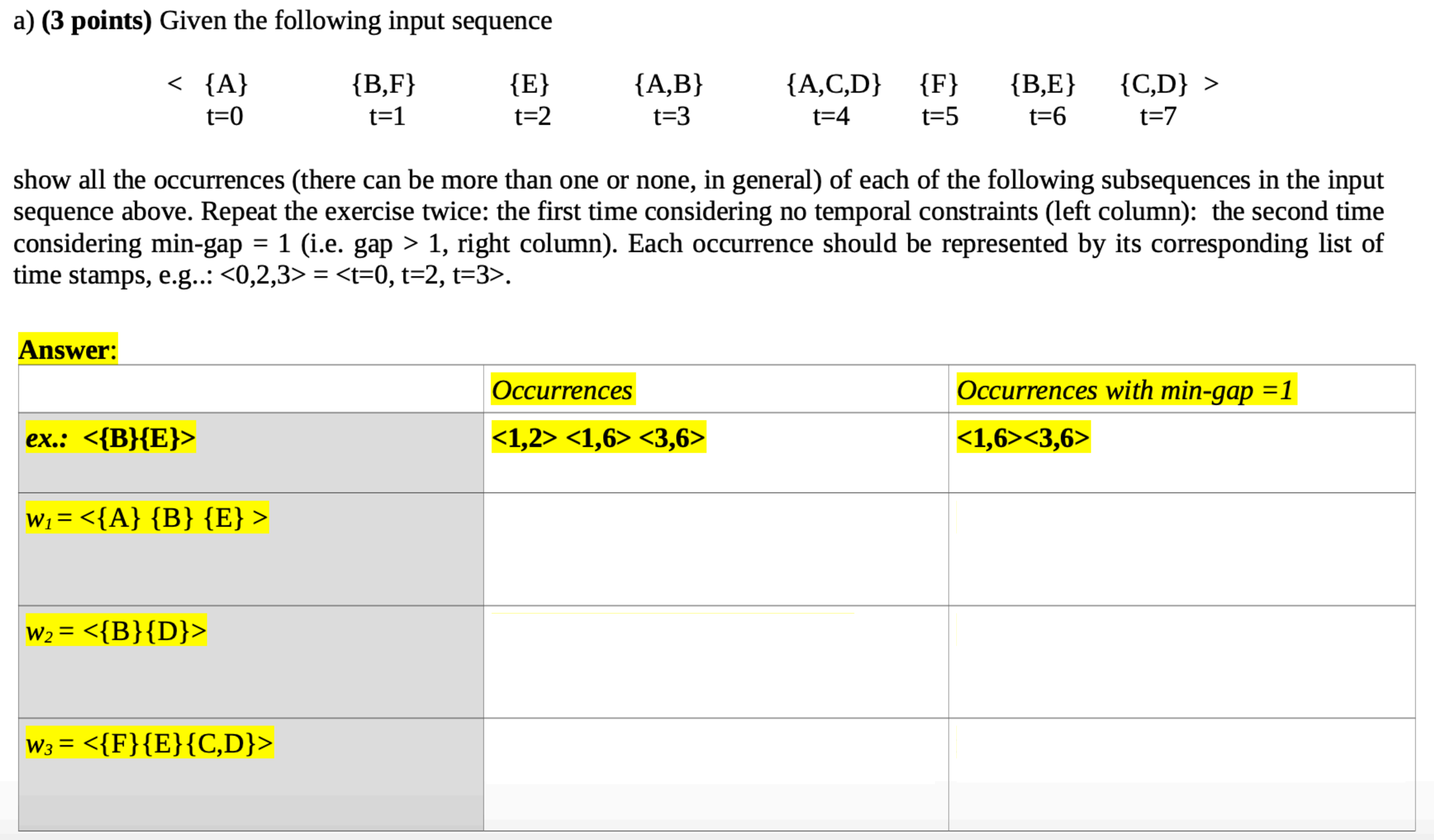
a) **(3 points)** Given the following input sequence

<	{A}	{B,F}	{E}	{A,B}	{A,C,D}	{F}	{B,E}	{C,D} >
	t=0	t=1	t=2	t=3	t=4	t=5	t=6	t=7

show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column): the second time considering min-gap = 1 (i.e. gap > 1, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g..: <0,2,3> = <t=0, t=2, t=3>.

	Occurrences	*Occurrences with min-gap =1*
***ex.:* <{B}{E}>**	**<1,2> <1,6> <3,6>**	**<1,6><3,6>**
w_1 = <{A} {B} {E} >		
w_2 = <{B}{D}>		
w_3 = <{F}{E}{C,D}>		

Sequential Pattern – Exercise 1



a) **(3 points)** Given the following input sequence

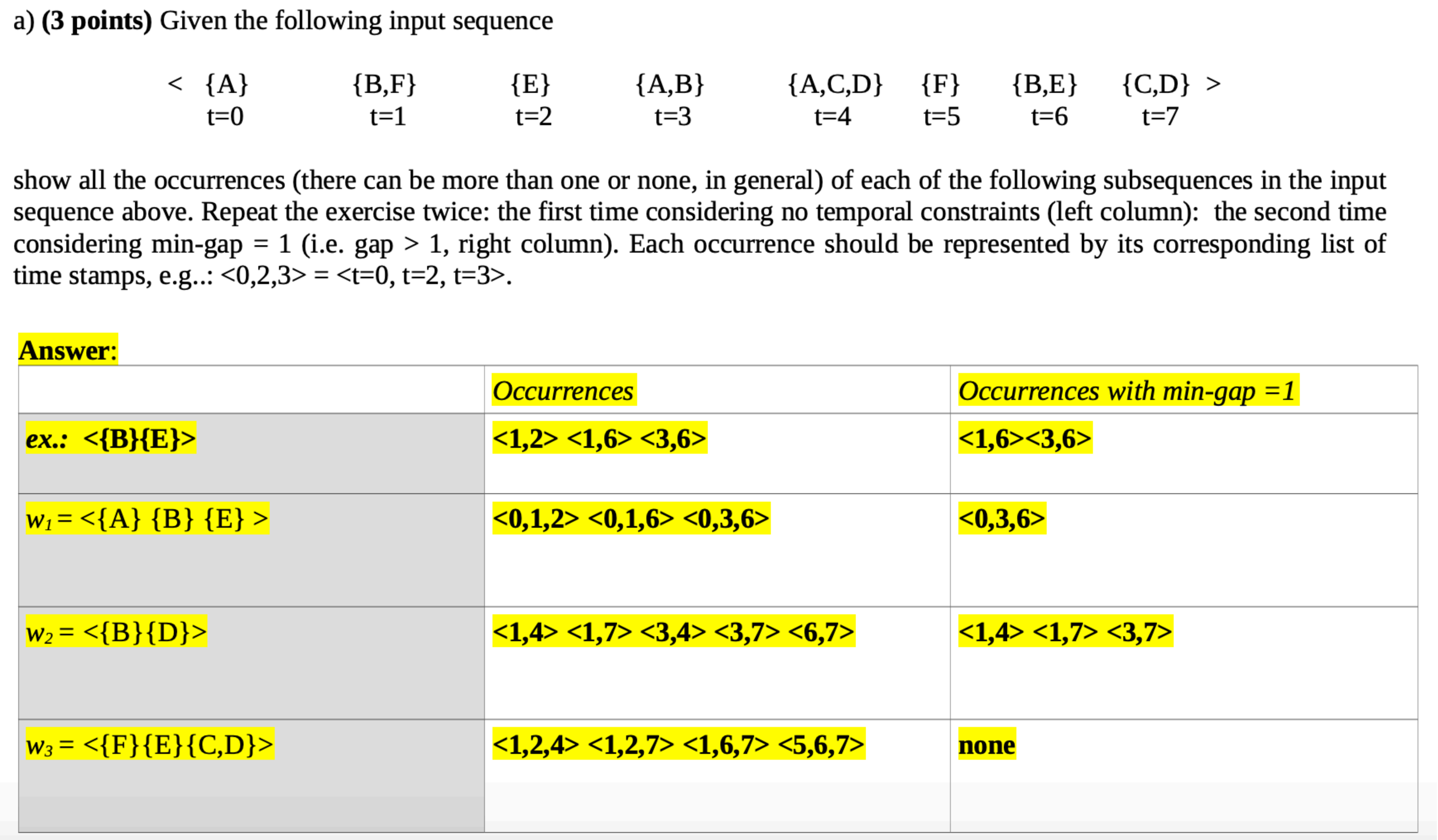
< {A}	{B,F}	{E}	{A,B}	{A,C,D}	{F}	{B,E}	{C,D} >
t=0	t=1	t=2	t=3	t=4	t=5	t=6	t=7

show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column): the second time considering min-gap = 1 (i.e. gap > 1, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g..: <0,2,3> = <t=0, t=2, t=3>.

Answer:

	Occurrences	*Occurrences with min-gap =1*
ex.: **<{B}{E}>**	**<1,2> <1,6> <3,6>**	**<1,6><3,6>**
w_1 = <{A} {B} {E} >		
w_2 = <{B}{D}>		
w_3 = <{F}{E}{C,D}>		

Sequential Pattern – Exercise 1 – Solution



a) **(3 points)** Given the following input sequence

<	{A}	{B,F}	{E}	{A,B}	{A,C,D}	{F}	{B,E}	{C,D}	>
	t=0	t=1	t=2	t=3	t=4	t=5	t=6	t=7	

show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column): the second time considering min-gap = 1 (i.e. gap > 1, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g..: <0,2,3> = <t=0, t=2, t=3>.

Answer:

	Occurrences	*Occurrences with min-gap =1*
ex.: **<{B}{E}>**	**<1,2> <1,6> <3,6>**	**<1,6><3,6>**
w_1 = <{A} {B} {E} >	**<0,1,2> <0,1,6> <0,3,6>**	**<0,3,6>**
w_2 = <{B}{D}>	**<1,4> <1,7> <3,4> <3,7> <6,7>**	**<1,4> <1,7> <3,7>**
w_3 = <{F}{E}{C,D}>	**<1,2,4> <1,2,7> <1,6,7> <5,6,7>**	**none**

Sequential Pattern – Exercise 2

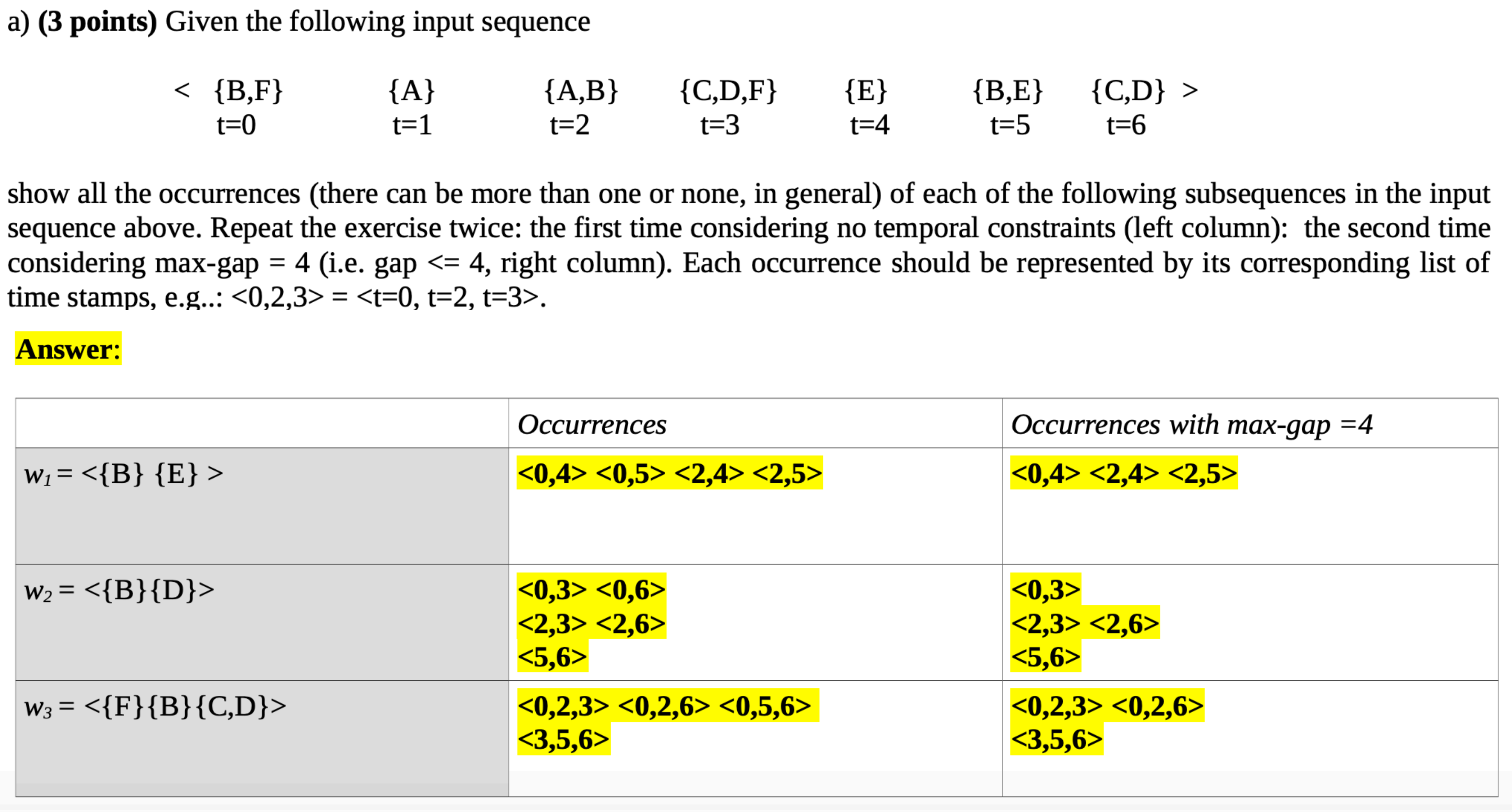
a) **(3 points)** Given the following input sequence

<	{B,F}	{A}	{A,B}	{C,D,F}	{E}	{B,E}	{C,D}	>
	t=0	t=1	t=2	t=3	t=4	t=5	t=6	

show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column): the second time considering max-gap = 4 (i.e. gap <= 4, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g..: <0,2,3> = <t=0, t=2, t=3>.

	Occurrences	*Occurrences with max-gap =4*
w_1 = <{B} {E} >		
w_2 = <{B}{D}>		
w_3 = <{F}{B}{C,D}>		

Sequential Pattern – Exercise 2 – Solution



a) **(3 points)** Given the following input sequence

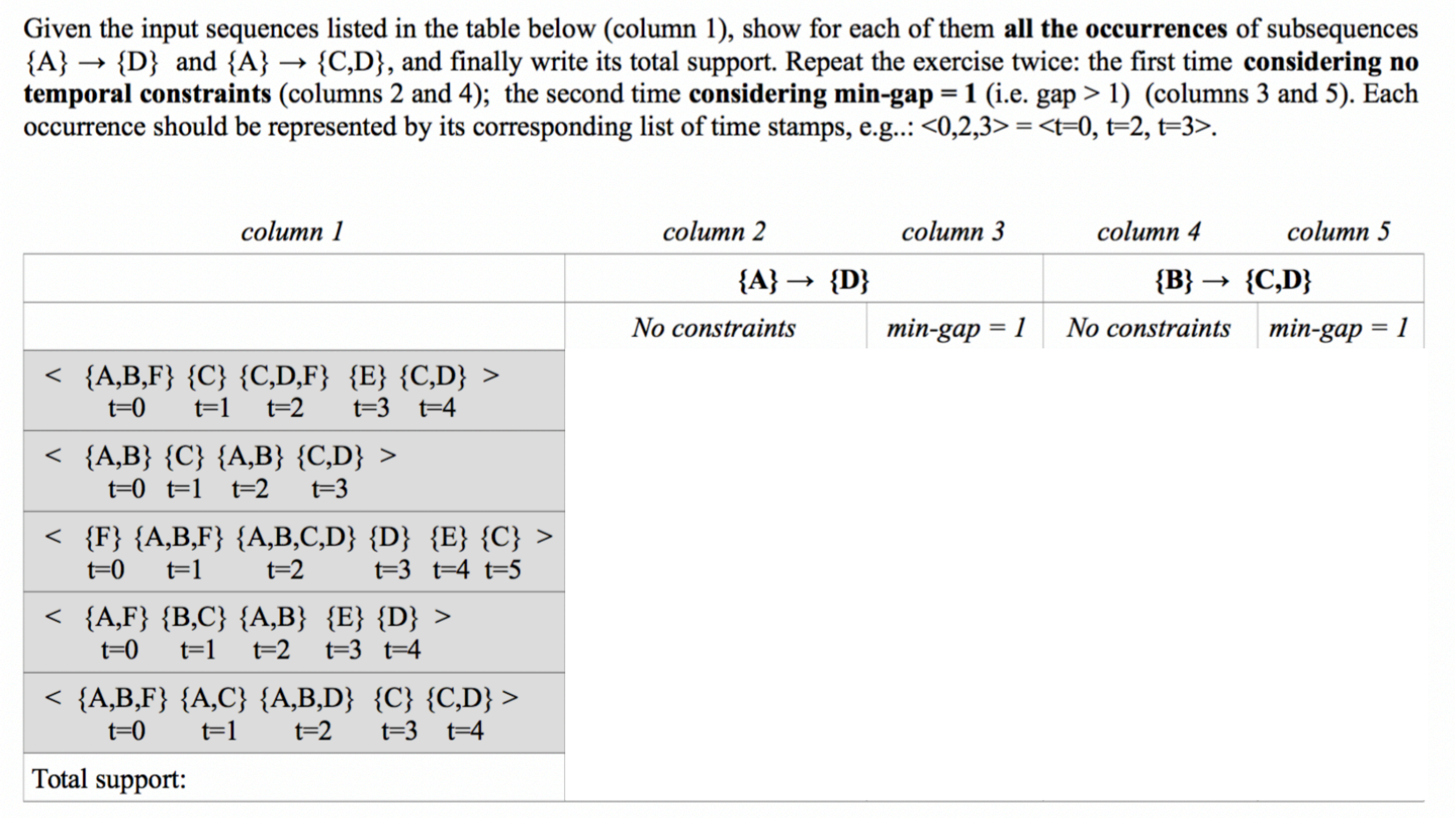
< {B,F} {A} {A,B} {C,D,F} {E} {B,E} {C,D} >
t=0 t=1 t=2 t=3 t=4 t=5 t=6

show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column): the second time considering max-gap = 4 (i.e. gap <= 4, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g..: <0,2,3> = <t=0, t=2, t=3>.

Answer:

	Occurrences	*Occurrences with max-gap =4*
w_1 = <{B} {E} >	**<0,4> <0,5> <2,4> <2,5>**	**<0,4> <2,4> <2,5>**
w_2 = <{B}{D}>	**<0,3> <0,6> <2,3> <2,6> <5,6>**	**<0,3> <2,3> <2,6> <5,6>**
w_3 = <{F}{B}{C,D}>	**<0,2,3> <0,2,6> <0,5,6> <3,5,6>**	**<0,2,3> <0,2,6> <3,5,6>**

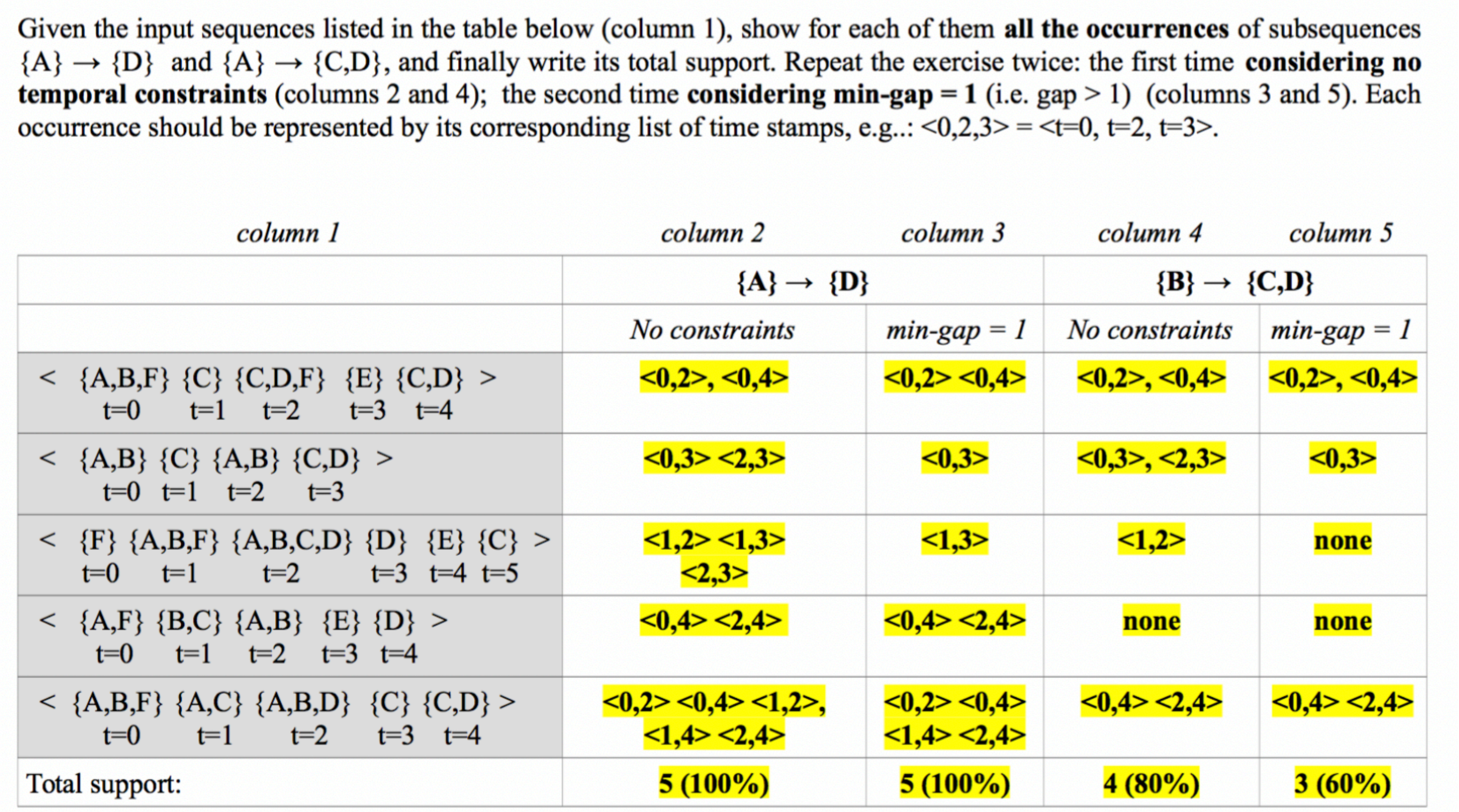
Sequential Pattern – Exercise 3



Given the input sequences listed in the table below (column 1), show for each of them **all the occurrences** of subsequences {A} → {D} and {A} → {C,D}, and finally write its total support. Repeat the exercise twice: the first time **considering no temporal constraints** (columns 2 and 4); the second time **considering min-gap = 1** (i.e. gap > 1) (columns 3 and 5). Each occurrence should be represented by its corresponding list of time stamps, e.g..: <0,2,3> = <t=0, t=2, t=3>.

column 1	*column 2*	*column 3*	*column 4*	*column 5*
	{A} → {D}		**{B} → {C,D}**	
	No constraints	*min-gap = 1*	*No constraints*	*min-gap = 1*
< {A,B,F} {C} {C,D,F} {E} {C,D} > t=0 t=1 t=2 t=3 t=4				
< {A,B} {C} {A,B} {C,D} > t=0 t=1 t=2 t=3				
< {F} {A,B,F} {A,B,C,D} {D} {E} {C} > t=0 t=1 t=2 t=3 t=4 t=5				
< {A,F} {B,C} {A,B} {E} {D} > t=0 t=1 t=2 t=3 t=4				
< {A,B,F} {A,C} {A,B,D} {C} {C,D} > t=0 t=1 t=2 t=3 t=4				
Total support:				

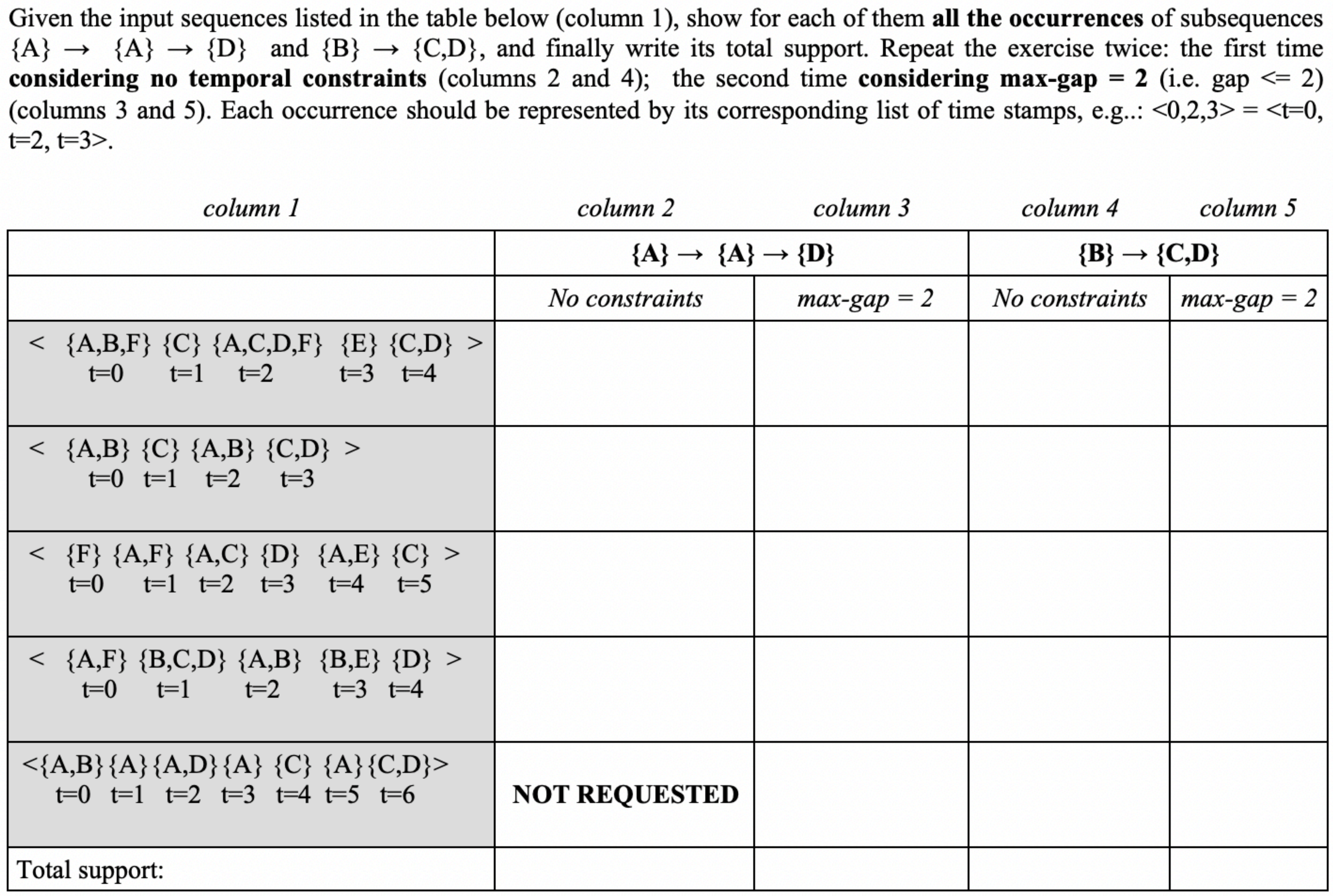
Sequential Pattern – Exercise 3 – Solution



Given the input sequences listed in the table below (column 1), show for each of them **all the occurrences** of subsequences {A} → {D} and {A} → {C,D}, and finally write its total support. Repeat the exercise twice: the first time **considering no temporal constraints** (columns 2 and 4); the second time **considering min-gap = 1** (i.e. gap > 1) (columns 3 and 5). Each occurrence should be represented by its corresponding list of time stamps, e.g..: <0,2,3> = <t=0, t=2, t=3>.

column 1	*column 2*	*column 3*	*column 4*	*column 5*
	{A} → {D}		**{B} → {C,D}**	
	No constraints	*min-gap = 1*	*No constraints*	*min-gap = 1*
< {A,B,F} {C} {C,D,F} {E} {C,D} > t=0 t=1 t=2 t=3 t=4	**<0,2>, <0,4>**	**<0,2> <0,4>**	**<0,2>, <0,4>**	**<0,2>, <0,4>**
< {A,B} {C} {A,B} {C,D} > t=0 t=1 t=2 t=3	**<0,3> <2,3>**	**<0,3>**	**<0,3>, <2,3>**	**<0,3>**
< {F} {A,B,F} {A,B,C,D} {D} {E} {C} > t=0 t=1 t=2 t=3 t=4 t=5	**<1,2> <1,3> <2,3>**	**<1,3>**	**<1,2>**	**none**
< {A,F} {B,C} {A,B} {E} {D} > t=0 t=1 t=2 t=3 t=4	**<0,4> <2,4>**	**<0,4> <2,4>**	**none**	**none**
< {A,B,F} {A,C} {A,B,D} {C} {C,D} > t=0 t=1 t=2 t=3 t=4	**<0,2> <0,4> <1,2>, <1,4> <2,4>**	**<0,2> <0,4> <1,4> <2,4>**	**<0,4> <2,4>**	**<0,4> <2,4>**
Total support:	**5 (100%)**	**5 (100%)**	**4 (80%)**	**3 (60%)**

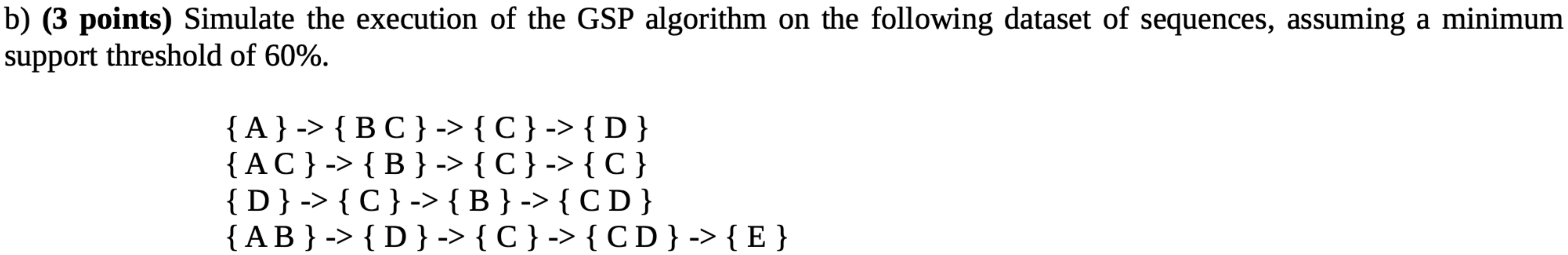
Sequential Pattern – Exercise 4



Given the input sequences listed in the table below (column 1), show for each of them **all the occurrences** of subsequences {A} → {A} → {D} and {B} → {C,D}, and finally write its total support. Repeat the exercise twice: the first time **considering no temporal constraints** (columns 2 and 4); the second time **considering max-gap = 2** (i.e. gap <= 2) (columns 3 and 5). Each occurrence should be represented by its corresponding list of time stamps, e.g..: <0,2,3> = <t=0, t=2, t=3>.

column 1	*column 2*	*column 3*	*column 4*	*column 5*
	{A} → {A} → {D}		**{B} → {C,D}**	
	No constraints	*max-gap = 2*	*No constraints*	*max-gap = 2*
< {A,B,F} {C} {A,C,D,F} {E} {C,D} > t=0 t=1 t=2 t=3 t=4				
< {A,B} {C} {A,B} {C,D} > t=0 t=1 t=2 t=3				
< {F} {A,F} {A,C} {D} {A,E} {C} > t=0 t=1 t=2 t=3 t=4 t=5				
< {A,F} {B,C,D} {A,B} {B,E} {D} > t=0 t=1 t=2 t=3 t=4				
<{A,B}{A}{A,D}{A} {C} {A}{C,D}> t=0 t=1 t=2 t=3 t=4 t=5 t=6	**NOT REQUESTED**			
Total support:				

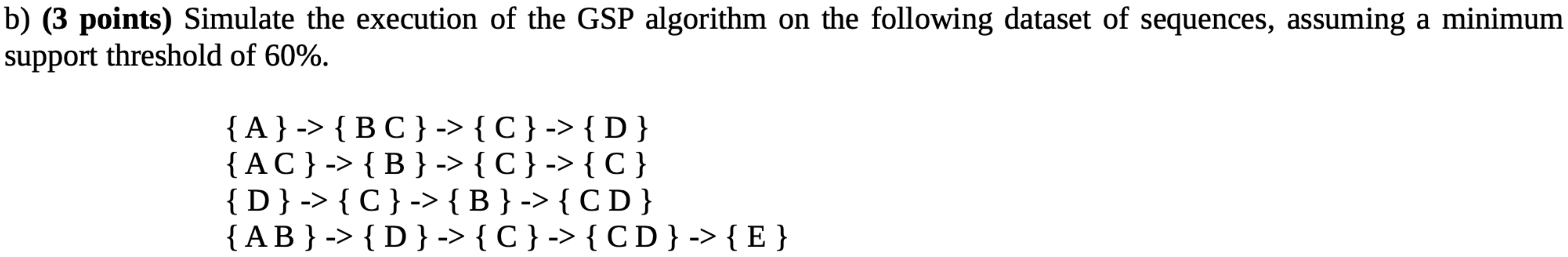
GSP – Exercise 1



b) **(3 points)** Simulate the execution of the GSP algorithm on the following dataset of sequences, assuming a minimum support threshold of 60%.

{ A } -> { B C } -> { C } -> { D }
{ A C } -> { B } -> { C } -> { C }
{ D } -> { C } -> { B } -> { C D }
{ A B } -> { D } -> { C } -> { C D } -> { E }

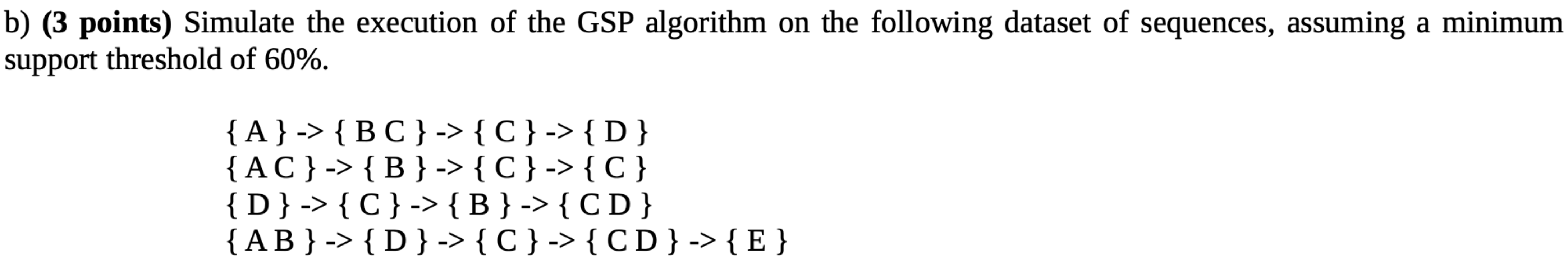
GSP – Exercise 1 – Solution



b) **(3 points)** Simulate the execution of the GSP algorithm on the following dataset of sequences, assuming a minimum support threshold of 60%.

{ A } -> { B C } -> { C } -> { D }
{ A C } -> { B } -> { C } -> { C }
{ D } -> { C } -> { B } -> { C D }
{ A B } -> { D } -> { C } -> { C D } -> { E }

GSP – Exercise 1 – Solution



b) **(3 points)** Simulate the execution of the GSP algorithm on the following dataset of sequences, assuming a minimum support threshold of 60%.

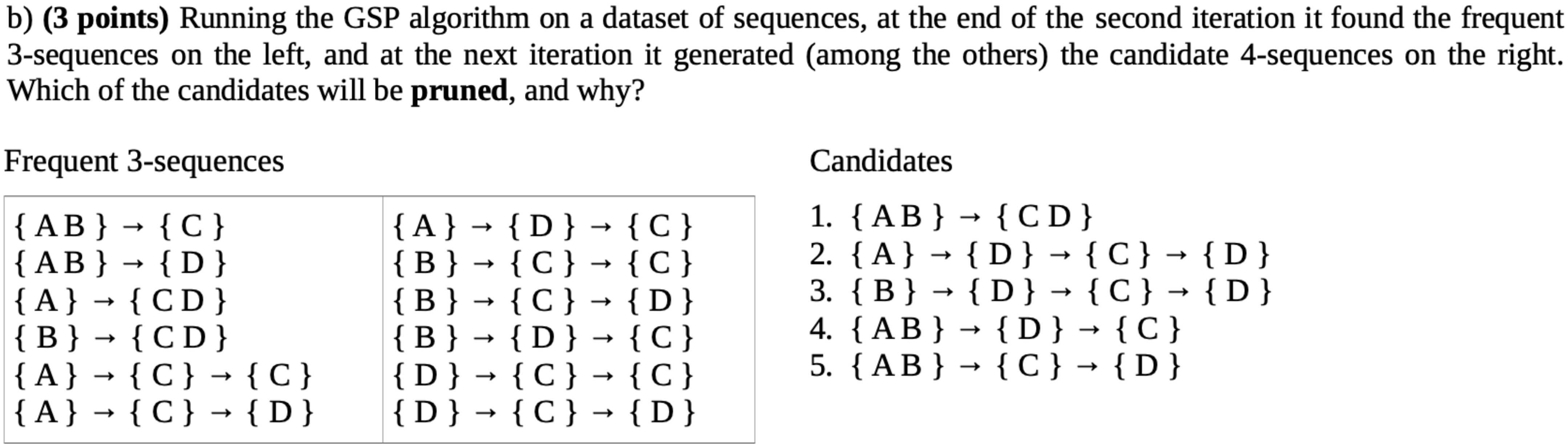
{ A } -> { B C } -> { C } -> { D }
{ A C } -> { B } -> { C } -> { C }
{ D } -> { C } -> { B } -> { C D }
{ A B } -> { D } -> { C } -> { C D } -> { E }

{ A }
{ B }
{ C }
{ D }
~~{ BC }~~
~~{ AC }~~
~~{ CD }~~
~~{ AB }~~

~~{ A } -> { B }~~
{ A } -> { C }
~~{ A } -> { D }~~
{ B } -> { C }
{ B } -> { D }
~~{ C } -> { B }~~
{ C } -> { C }
{ C } -> { D }
~~{ D } -> { B }~~
~~{ D } -> { C }~~
~~{ D } -> { D }~~

{ A } -> { C } -> { C }
~~{ A } -> { C } -> { D }~~ (pruning)
~~{ B } -> { C } -> { D }~~
~~{ B } -> { C } -> { C }~~
~~{ C } -> { C } -> { C }~~

GSP – Exercise 2



b) **(3 points)** Running the GSP algorithm on a dataset of sequences, at the end of the second iteration it found the frequent 3-sequences on the left, and at the next iteration it generated (among the others) the candidate 4-sequences on the right. Which of the candidates will be **pruned**, and why?

Frequent 3-sequences

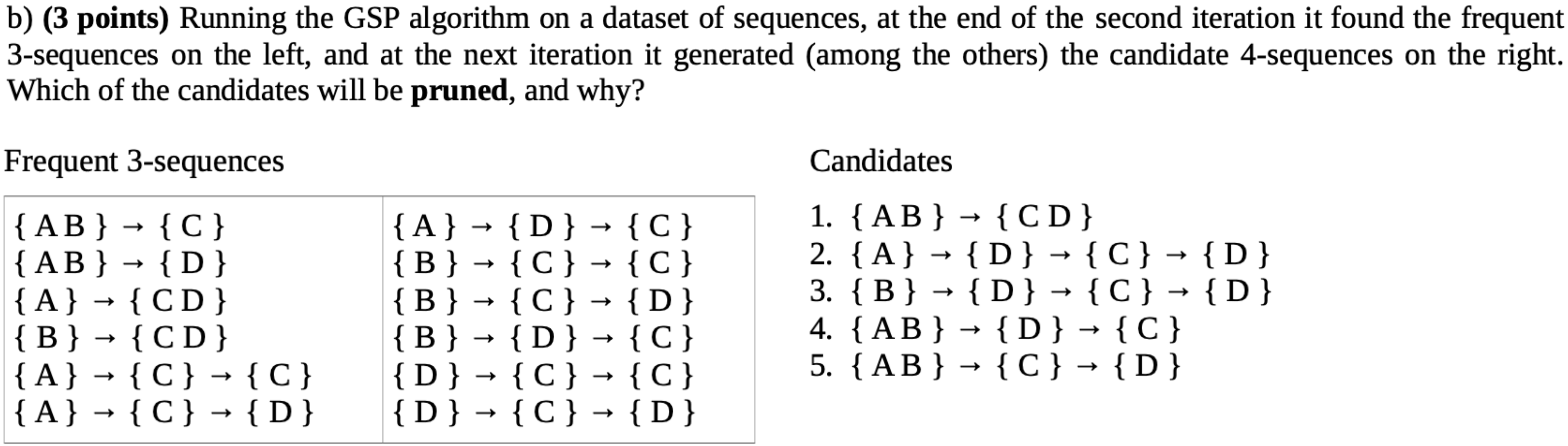
{ A B } → { C }
{ A B } → { D }
{ A } → { C D }
{ B } → { C D }
{ A } → { C } → { C }
{ A } → { C } → { D }

{ A } → { D } → { C }
{ B } → { C } → { C }
{ B } → { C } → { D }
{ B } → { D } → { C }
{ D } → { C } → { C }
{ D } → { C } → { D }

Candidates

1. { A B } → { C D }
2. { A } → { D } → { C } → { D }
3. { B } → { D } → { C } → { D }
4. { A B } → { D } → { C }
5. { A B } → { C } → { D }

GSP – Exercise 2 – Solution



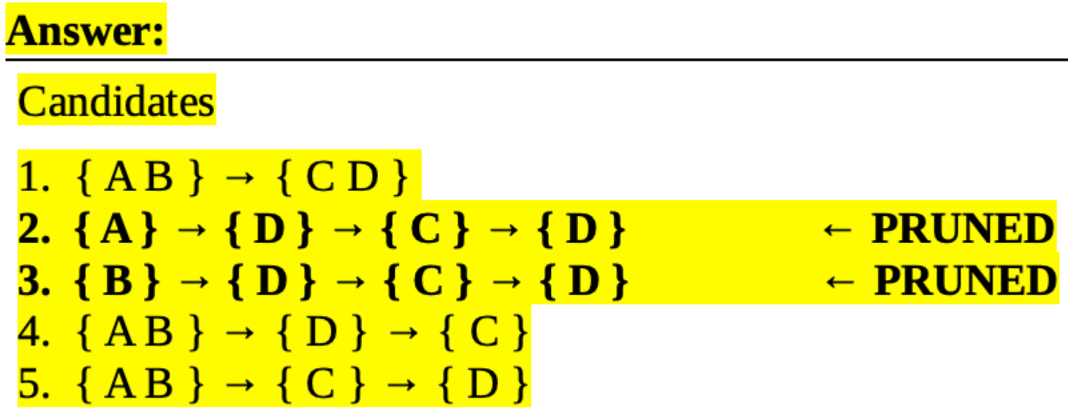
b) **(3 points)** Running the GSP algorithm on a dataset of sequences, at the end of the second iteration it found the frequent 3-sequences on the left, and at the next iteration it generated (among the others) the candidate 4-sequences on the right. Which of the candidates will be **pruned**, and why?

Frequent 3-sequences

{ A B } → { C }
{ A B } → { D }
{ A } → { C D }
{ B } → { C D }
{ A } → { C } → { C }
{ A } → { C } → { D }
{ A } → { D } → { C }
{ B } → { C } → { C }
{ B } → { C } → { D }
{ B } → { D } → { C }
{ D } → { C } → { C }
{ D } → { C } → { D }

Candidates

1. { A B } → { C D }
2. { A } → { D } → { C } → { D }
3. { B } → { D } → { C } → { D }
4. { A B } → { D } → { C }
5. { A B } → { C } → { D }



Answer:

Candidates

1. { A B } → { C D }
2. **{ A } → { D } → { C } → { D }** ← **PRUNED**
3. **{ B } → { D } → { C } → { D }** ← **PRUNED**
4. { A B } → { D } → { C }
5. { A B } → { C } → { D }

Missing from frequent 3-sequences

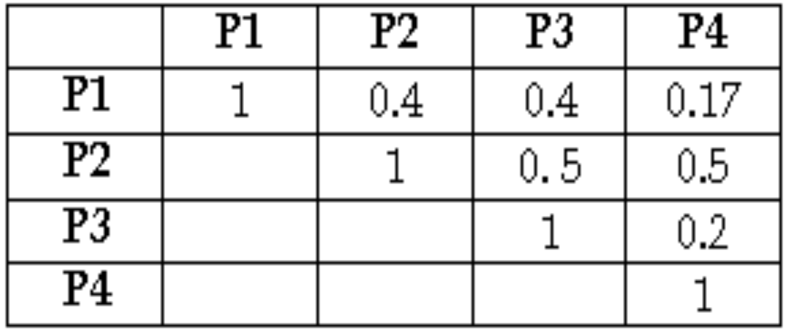
- A -> D -> D
- B -> D -> D

Transactional Clustering

Rock – Exercise 1

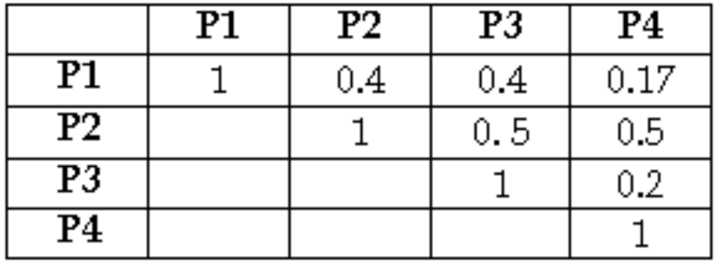
- Suppose we have four verses contains some subjects , as follows:
- P1={ judgment, faith, prayer, fair}
- P2={ fasting, faith, prayer}
- P3={ fair, fasting, faith}
- P4={ fasting, prayer, pilgrimage}
- **the similarity threshold = 0.3, and number of required cluster is 2.**

Using Jaccard coefficient as a similarity measure, we obtain the following similarity table



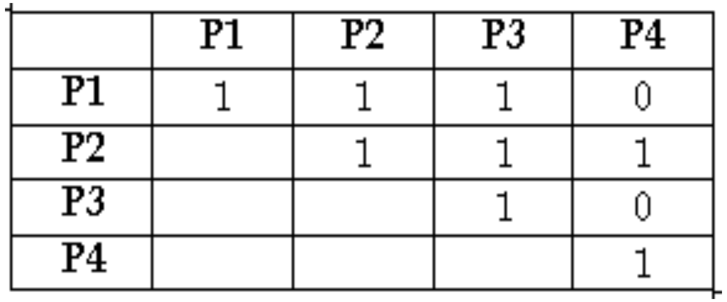
	P1	P2	P3	P4
P1	1	0.4	0.4	0.17
P2		1	0. 5	0.5
P3			1	0.2
P4				1

Rock – Exercise 1



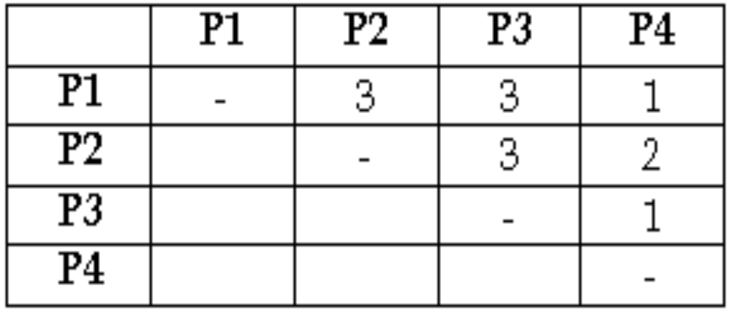
	P1	P2	P3	P4
P1	1	0.4	0.4	0.17
P2		1	0. 5	0.5
P3			1	0.2
P4				1

- Since we have a similarity threshold equal to 0.3, then we derive the adjacency table: →



	P1	P2	P3	P4
P1	1	1	1	0
P2		1	1	1
P3			1	0
P4				1

- By multiplying the adjacency table with itself, we derive the following table which shows the number of links (or common neighbors): →



	P1	P2	P3	P4
P1	-	3	3	1
P2		-	3	2
P3			-	1
P4				-

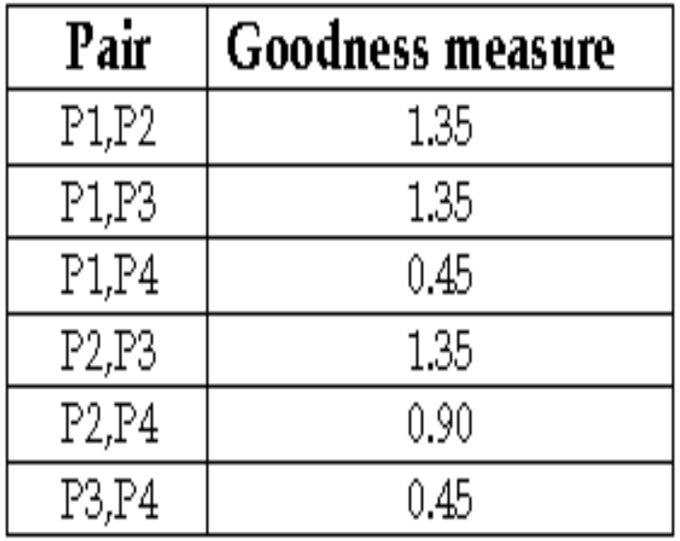
Rock – Exercise 1

- we compute the goodness measure for all adjacent points ,assuming that

$$g(P_i, P_j) = \frac{link[P_i, P_j]}{(n+m)^{1+2f(\theta)} - n^{1+2f(\theta)} - m^{1+2f(\theta)}}$$

- $f(\theta) = 1-\theta \ / \ 1+\theta$ = 1-0.3 /1+0.3 = 0.54

- we obtain the following table➔



Pair	Goodness measure
P1,P2	1.35
P1,P3	1.35
P1,P4	0.45
P2,P3	1.35
P2,P4	0.90
P3,P4	0.45

- we have an equal goodness measure for merging ((P1,P2), (P2,P3), (P3,P1))

Rock – Exercise 1

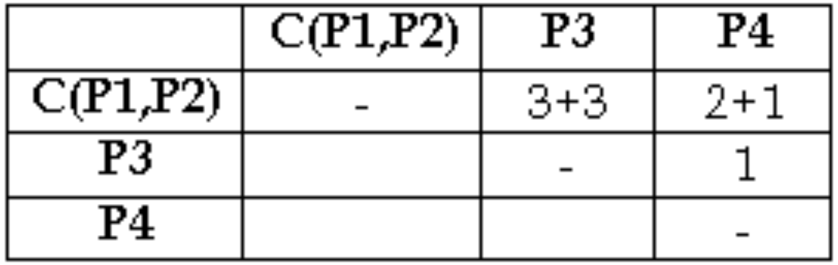
- Now, we start the hierarchical algorithm by merging, say P1 and P2.

- A new cluster (let's call it C(P1,P2)) is formed.

- It should be noted that for some other hierarchical clustering techniques, we will not start the clustering process by merging P1 and P2, since Sim(P1,P2) = 0.4,which is not the highest. But, ROCK uses the number of links as the similarity measure rather than distance.

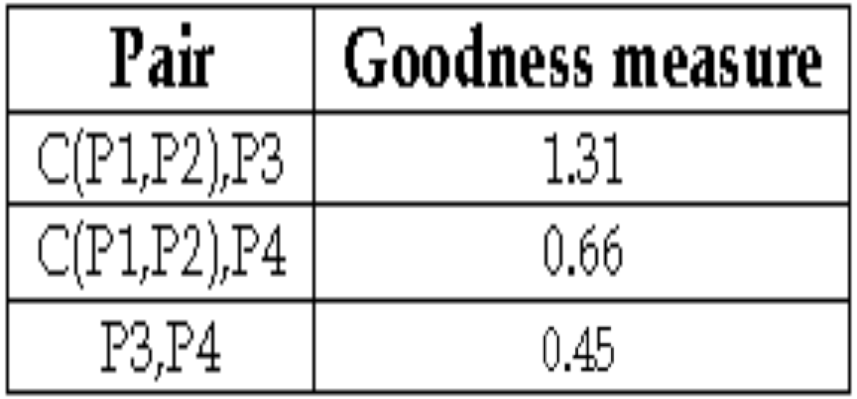
Rock – Exercise 1

- Now, after merging P1 and P2, we have only three clusters. The following table shows the number of common neighbors for these clusters:→



	C(P1,P2)	P3	P4
C(P1,P2)	-	3+3	2+1
P3		-	1
P4			-

- Then we can obtain the following goodness measures for all adjacent clusters:→



Pair	Goodness measure
C(P1,P2),P3	1.31
C(P1,P2),P4	0.66
P3,P4	0.45

Rock – Exercise 1

- Since the number of required clusters is 2, then we finish the clustering algorithm by merging C(P1,P2) and P3, obtaining a new cluster C(P1,P2,P3) which contains {P1,P2,P3} leaving P4 alone in a separate cluster.

Rock – Exercise 2

- Given the following similarity matrix find the clustering result knowing that the similarity threshold = 0.4, and number of required cluster is 2.

	p1	p2	p3	p4	p5
p1	1	0.7	0.2	0.5	0.5
p2		1	0.6	0.8	0.1
p3			1	0.5	0.4
p4				1	0.3
p5					1

Rock – Exercise 2 – Solution

	p1	p2	p3	p4	p5
p1	1	0.7	0.2	0.5	0.5
p2		1	0.6	0.8	0.1
p3			1	0.5	0.4
p4				1	0.3
p5					1

	p1	p2	p3	p4	p5
p1	1	1	0	1	1
p2	1	1	1	1	0
p3	0	1	1	1	1
p4	1	1	1	1	0
p5	1	0	1	0	1

Rock – Exercise 2 – Solution

	p1	p2	p3	p4	p5
p1	1	1	0	1	1
p2	1	1	1	1	0
p3	0	1	1	1	1
p4	1	1	1	1	0
p5	1	0	1	0	1

	p1	p2	p3	p4	p5
p1	-	3	3	3	2
p2		-	3	4	2
p3			-	3	2
p4				-	2
p5					-

Rock – Exercise 2 – Solution

- $f(\theta) = 1-\theta \ / \ 1+\theta$ = 1-0.4 /1+0.4 = 0.43
- 1 + 2 $f(\theta) = 1.86$

$$g(P_i, P_j) = \frac{link[P_i, P_j]}{(n+m)^{1+2f(\theta)} - n^{1+2f(\theta)} - m^{1+2f(\theta)}}$$

	p1	p2	p3	p4	p5
p1	-	3	3	3	2
p2		-	3	4	2
p3			-	3	2
p4				-	2
p5					-

	p1	p2	p3	p4	p5
p1	-	1.84	1.84	1.84	1.22
p2		-	1.84	2.45	1.22
p3			-	1.84	1.22
p4				-	1.84
p5					-

Rock – Exercise 2 – Solution

- $f(\theta)$ = $1-\theta$ / $1+\theta$ = 1-0.4 /1+0.4 = 0.43
- 1 + 2 $f(\theta)$ = *1.86*

$$g(P_i,P_j)=\frac{link[P_i,P_j]}{(n+m)^{1+2f(\theta)}-n^{1+2f(\theta)}-m^{1+2f(\theta)}}$$

	p1	p2	p3	p4	p5
p1	-	3	3	3	2
p2		-	3	4	2
p3			-	3	2
p4				-	2
p5					-

	p1	p2p4	p3	p5
p1	-	6	3	2
p2p4		-	6	4
p3			-	2
p5				-

	p1	p2p4	p3	p5
p1	-	1.94	1.84	1.22
p2p4		-	1.94	1.29
p3			-	1.22
p5				-

- *Final Clusters: p1234 p5*

Clope Exercise 1

Split1:

- 4 transactions: abc, abc, ab, a
  - a: 4, b:3, c: 2 -> sol: S=9; W=3; H=9/3=3; H/W=1
- 3 transactions: def, de, de
  - d: 3, e:3, f: 1 -> sol: S=7; W=3; H=7/3=2.33; H/W=0.77

$$Profit_r(\mathbf{C}) = \frac{\sum_{i=1}^{k} \frac{S(C_i)}{W(C_i)^r} \times |C_i|}{\sum_{i=1}^{k} |C_i|}$$

Split2:

- 2 transactions: abcd, ab
  - a: 2, b:2, c: 1, d:1 -> sol: S=6; W=4; H=6/4=1.5; H/W=0.37
- 2 transactions: ec, ec
  - e:2, c: 2 -> sol: S=4; W=2; H=4/2=2; H/W=1

Split1 is the best clustering considering r=2

Profit(Split1) = (9/3^2 * 4 + 7/3^2 * 3) /7 = 0.90

Profit(Split2) = (6/4^2 * 2 + 4/2^2 * 2) /4 = 0.16