Question 1: Mobile Data Broadcasting

1.1 Broadcast schedule *(15pts)*

The pages are grouped by access frequency to form disks. Access frequencies must be computed by taking into account the interest of customer and number of pages in each categories.

Category	Number of pages	% of customers	Access freq. per page	Disks
News	50	45	0.009	A
Sports	75	30	0.004	B
TV/Radio	100	10	0.001	C
Travel	30	5	0.0016666	D
Finance	50	5	0.001	C
Entertainment	30	5	0.0016666	D

a) The disks and their frequencies *(5pts)*

Disks	Optimal frequencies	Number of chunks (cmax=6)
A: pages 100-149	3	2
B: pages 200-274	2	3
C: 300-399 and 500-549	1	6
D: 400-429 and 600-629	1	6

b) The list of chunks *(5pts)*

According to the table above, the 4 disks are divide in chunks of same size.
A1: 100-124, A2: 125-149,B1: 200-224,B2: 225-249,B3: 250-274,C1: 300-324,C2: 325-349,C3: 350-374, C4: 375-399, C5: 500-524, C6: 525-549, D1: 400-409, D2: 410-419, D3: 420-429, D4: 600-609, D5: 610-619, D6: 620-629

c) The optimal schedule *(5pts)*

A1 B1 C1 D1 A2 B2 C2 D2 A1 B3 C3 D3 A2 B1 C4 D4 A1 B2 C5 D5 A2 B3 C6 D6

1.2 Quizz *(10 pts)*

a) The largest latency is with fanout=2 *(5pts)*
-> Latency (fanout=2):|I| + N + C = 254+256+ C = 510 + C
Latency (fanout=4): 84+256 + C = 340 + C
Latency (fanout=16): 16+256 + C = 272 + C

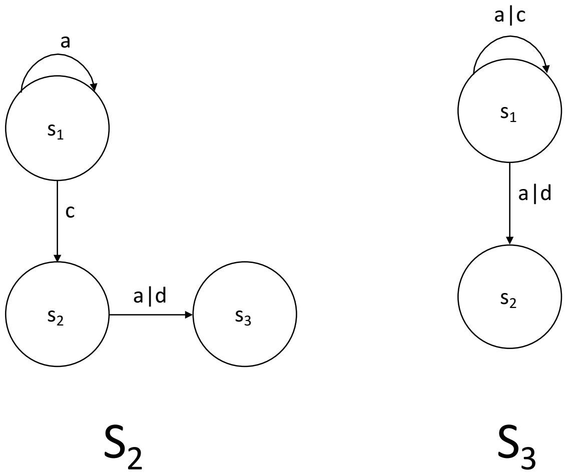
b) The shortest tuning time is with fanout=4 *(5pts)*
Tuning time (fanout=2): = (n+1)/2*$\log_n$(N) = 1.5 * $\log_2$(256) = 12
->Tuning time (fanout=4): 2.5 * $\log_4$(256) = 10
Tuning time (fanout=16): 8.5 * $\log_{16}$(256) = 17
Or when using the approximation formula (24, 20 34)

Question 2: Graph Databases

Question 2.1 (3 pts) Answer *c*

Question 2.2 (3 pts)

Solution: One possibility to get S_2 is adding a label d from s_2 to s_3:



Question 2.3 (4 pts) The answers below give the classification for the maximal simulation if we add the label d from s_2 to s_3

class	instances
s_1	d_1, d_2, d_4, d_5
s_2	d_3, d_6
s_3	d_7

Question 2.4 (5 pts) Solution:

The data guide is equivalent to the data graph D itself.

Question 2.5 (5 pts)

Solution:

Check S_3 in Figure above

Question 2.6 (5 pts)

Answer *c*

Q3.1 (10 points)

football	Brazil	Germany	Aggregation
1,0.85	1,0.78	1,0.65	0.85
3,0.8	3,0.96	3,0.78	0.96
4,0.78	4,0.96	4,0.85	0.96
5,0.3	5,0.91	5,0.75	0.91
6,0.65	6,0.79	6,0.72	0.79
7,0.71	7,0.85	7,0.61	0.85
8,0.73	8,0.6	8,0.94	0.94
10,0.6	10,0.31	10,0.91	0.91
11,0.63	11,0.4	11,0.37	0.63
12,0.49	12,0.72	12,0.9	0.9
-1point	- 1point	-1point	-1point

Top-3: document 3,4 and 8 - 2 points

Scan phase:

Totally, 4 blocks of items are transferred from the three posting-list nodes to the requesting node. -1point

4*2*3=24 items are sent through the network. But, the requesting node processes until the 7-th item of the posting list, since it collects 3 document-ids that appear in all the three posting lists.

Random access phase: the red items in above table represent the ones should be retrieved by random access.

In "football" posting list, document 10 is already transferred to the requesting node via the block data access in the scan phase. -1point

In "Germany" posting list, document 1 is already transferred to the requesting node in the scan phase. -1point

In total, there are 2+3+2=7 items transferred through the network via random access. -1point

3.2 (15 points)

The items in the cache of the requesting node are the same as above table. The aggregation results are shown in the following table.

Top-2: 3, 4 -2 point

football	Brazil	Germany	Belgium	Aggregation
1	1	1	1, 0.81	0.85
3	3	3	3,0.95	0.96

4	4	4	4,0.75	0.96
5	5	5	5,0.89	0.91
6	6	6	6,0.66	0.79
7	7	7	7,0.88	0.88
8	8	8	8,0.83	0.94
10	10	10	10,0.73	0.91
12	12	12	12,0.43	0.9
-1point	- 1point	-1point	-1point	-2point

In the scan phase, the requesting node scans to the 6-th items, 3 blocks of the "Belgium" posting list. -2point

We can see that the items that are needed to retrieve from the other three posting lists are all in the local cache. -3point

And, the red items in the "Belgium" column are the ones that need random access. In summary, the number of items that are transferred through the network is 3*2+3=9. -2point

Q4: Recommender systems

Q4.1

We first compute the average ratings of each users:

U1	U2	U3	U4
4	2	3	4

We then create a version of the ratings table shifted by the average rating of each users. **Not necessary, but allows to see that U2 and U3 have a rating = 0 for page b therefore the rating of U1 for page b is already known: Its average rating (4).**

	U1	U2	U3	U4
a	1	0	1	0
b		0	0	0
c	1	1	-2	-1
d	-2		1	0
e	0	-1		1

We then compute the Pearson correlation between 2 users for the pairs of interest using only the pages that both users already rated. Given the imposed neighborhoods, only u1-u2, u1-u3 and u2-u4 are required (coincidence, they are nice numbers):

u1-u2	0.5
u1-u3	-0.5
u1-u4	Not needed
u2-u3	Not needed
u2-u4	-1
u3-u4	Not needed

We finally apply the user-based collaborative filtering formula giving the following ratings:

	U1	U2	U3	U4
a				
b	4			
c				
d		1.33333333		
e			3	

Q4.2

We first apply the simple formula given in the question to compute the similarity of sentiment between pages belonging to the same neighborhoods. We cut on some computation here as well as A and C ratings are known for every users, no need to compute their similarity.

a-b	0.75
b-c	0.25
d-e	1

We then compute the keyword similarity using the Jaccard similarity (relatively fast to do)

	union	intersection	Jaccard
a-b	4	8	0.5
b-c	1	10	0.1
d-e	2	8	0.25

We compute the final similarity between 2 pages as the product of the 2 previously computed similarities:

a-b	0.375	=3/8
b-c	0.025	=1/40
d-e	0.25	=1/4

We use this similarity as weights of the average to compute the final score for the pages (the similarities being between [0; 1], the average is rightly between [1; 5]).

	U1	U2	U3	U4
a				
b	5			
c				
d		1		
e			4	

NB: Due to simplification of the numbers, the rating of page B is the result of an average between 5 and 5 and therefore can be computed immediately. The rating of pages d and e are the results of an average of only one element and can also be computed immediately.

Q4.3

The pages are organized as a ring, therefore they all have the same hub and authority score. We can distinguish two cases:

In terms of probabilities

We can normalize the hub/authority vector so that the sum of their components equals 1 as would a probability distribution. **This way, each page have a hub/authority of 1/5**. Giving a **LinkedBasedRating of 1.8.**

Using a Euclidian normalization

We can also normalize the hub/authority vector using the Euclidian normalization (division by the length of the vector). **This way, each page have a hub/authority of 1/SQRT(5)**. Giving a **LinkedBasedRating of 2.79.**

Both answer were accepted.

Q4.4

Compute the weightless average of the 3 ratings computed previously using a linkedBasedRating of 1.8:

	U1	**U2**	**U3**	**U4**
a				
b	3.6			
c				
d		1.37		
e			2.93	

Or if used the alternative normalization 1/SQRT(5) in Q3 leading to a LinkedBasedRating of 2.79:

	U1	**U2**	**U3**	**U4**
a				
b	3.93			
c				
d		1.71		
e			3.26	