Question 1: Structured Overlay Network: Chord

This is the solution according to the lecture's definition of successor, meaning the successor(i) of a peer with key n is the smallest j such that there exist a peer at j and $j > i + n$.

Question 1.1 (12pt)

$\mathcal{H}(B) = 4$ can be found using fingers 1 and 2 of A.

$\mathcal{H}(G) = 18$ can be found using fingers 1 and 2 of F.

$\mathcal{H}(H) = 24$ can be found using finger 4 of F and finger 5 of C.

$\mathcal{H}(J) = 43$ can be found using finger 6 of D and finger 6 of J.

Question 1.2 (5pt)

The resource with key 17 is stored at node G.

I will query C (its finger 6), then C will query F (its finger 4), F will query G (its first finger) and G will return the value requested.

Question 1.3 (8pt)

i	d	peer A	peer B	peer C	peer D	peer E	peer F	peer G	peer H	peer I	peer J	peer M
1	1	B	C	D	E	F	G	H	I	J	**M**	**A**
2	2	C	C	D	F	F	H	H	I	J	**M**	**A**
3	4	C	D	E	F	G	H	H	I	J	**M**	**A**
4	8	E	F	F	H	H	I	I	I	**M**	A	**A**
5	16	H	H	H	I	I	I	I	J	A	A	**B**
6	32	I	J	J	J	**M**	**M**	A	A	C	E	**H**

Table 1: Fingers table at for all the peers after M joined.

Question 1: Structured Overlay Network: Chord

This is the solution according to the paper definition of successor, meaning the successor(i) of a peer with key n is the smallest j such that there exist a peer at j and $j \geq i + n$. Please note that the exercise was designed based on the other definition and some inconsistencies may appear in the routing table, but they have no impact on the solution.

Question 1.1 (12pt)

$\mathcal{H}(B) = 3$ or 5 (depending on where you start or if you consider all inequalities, it has no answer) can be found using fingers 1 and 2 of A, respectively 2 and 4 of B.

$\mathcal{H}(G) = 17$ can be found using fingers 1 and 2 of F.

$\mathcal{H}(H) = 23$ can be found using finger 4 of F and finger 5 of C.

$\mathcal{H}(J) = 43$ can be found using finger 6 of E and finger 6 of J.

All these results are independent, and don't influence each others.

Question 1.2 (5pt)

The resource with key 17 is stored at node G.

I will query C (its finger 6), then C will query F (its finger 4), F will query G (its first finger) and G will return the value requested.

Question 1.3 (8pt)

i	d	peer A	peer B	peer C	peer D	peer E	peer F	peer G	peer H	peer I	peer J	peer M
1	1	B	C	D	E	F	G	H	I	J	**M**	**A**
2	2	C	C	D	F	F	H	H	I	J	**M**	**A**
3	4	C	D	E	F	G	H	H	I	J	**M**	**A**
4	8	E	F	F	H	H	I	I	I	**M**	A	**A**
5	16	H	H	H	I	I	I	I	J	A	A	**A**
6	32	I	J	J	J	**M**	**M**	**M**	A	C	E	**H**

Table 1: Fingers table at for all the peers after M joined.

Distributed Information Systems
Final exam, winter semester 2012/2013
18 January 2013, 12.15 - 15.15

Solution for Question 2: XML Filtering
Given an XML document

```
<city>
    <name>A</name>
    <museum>
        <name>B</name>
        <ticket>X</ticket>
    </museum>
    <park>
        <name>C</name>
    </park>
</city>
```

and three XPath queries given in this order:

1. /city/name/park
2. //park
3. /city//museum/ticket

Question 2.1: (3 pts) Determine the path nodes for the 3 queries above (without using load balancing in generating the path nodes).

Answer:

1. **(1 pt)** Q_{11}(1,1,na), Q_{12}(2,2,1), Q_{13}(3,3,1)
2. **(1 pt)** Q_{21}(1, undet, na)
3. **(1 pt)** Q_{31}(1,1,na), Q_{32}(2, undet, undet or na), Q_{33}(3, undet, 1)

Question 2.2: (4 pts) Provide for all elements the candidate lists and waiting lists that are constructed for building the query index (without load balancing).

Answer:

- **(2 pts)** Candidate lists:
  city = $[Q_{11}, Q_{31}]$
  name = []
  museum = []
  ticket = []
  park = $[Q_{21}]$
- **(2 pts)** Waiting lists:
  city = []
  name = $[Q_{12}]$
  museum = $[Q_{32}]$
  ticket = $[Q_{33}]$
  park = $[Q_{13}]$

Question 2.3: (9 pts) Provide a detailed description of the processing steps of the XML document filtering engine when processing the XML document given above. Provide in a table for each step:

- the type of event processed (start/end/PCDATA),
- the action preformed for the event (match? promote/remove from candidate list?), and
- the candidate list after the processing of the event.

Answer:

See Q2.3.xlsx file.

Question 2.4: (3 pts) What are the matched and unmatched queries?

Answer:

- **(1 pt)** Q_1: unmatched query
- **(1 pt)** Q_2: matched query
- **(1 pt)** Q_3: matched query

Question 2.5: (6 pts) When using load balancing,

(a) determine the path nodes for the three queries above, and

(b) provide for all elements the candidate lists and waiting lists that are constructed!

Answer:

(a) **(3 pts)**

1. **(1 pt)** Q_{11}(1,1,na), Q_{12}(2,2,1), Q_{13}(3,3,1)
2. **(1 pt)** Q_{21}(1, undet, na)
3. **(1 pt)** Q_{31}(1,undet,undet or na,/city), Q_{32}(2, undet, 1)

(b) **(3 pts)**

- **(1.5 pts)** Candidate lists:

city	=	$[Q_{11}]$
name	=	[]
museum	=	$[Q_{31}]$
ticket	=	[]
park	=	$[Q_{21}]$

- **(1.5 pts)** Waiting lists:

city	=	[]
name	=	$[Q_{12}]$
museum	=	[]
ticket	=	$[Q_{32}]$
park	=	$[Q_{13}]$

Solution

Question 3.1.

a. We have : P(Y|X) = P(Y U X) / P (X) = 40/50 = 0.8

→ confidence ({X}→ {Y}) = 0.8

b. No, because the support (0.4) is less than the threshold (0.6).

Question 3.2.

1- Valid: Running an additional round will not result in any modifications of the assignments to clusters.
2- Valid: Running an additional round will not result in any modifications of the assignments to clusters.
3- Invalid: Running an extra round will make the points of C1 be in the cluster {C2}.
4- Valid: Running an additional round will not result in any modifications of the assignments to clusters.
5- Invalid: Running an extra round will make the points of C2 be in the cluster {C1}.

Question 3.3.

a. Entropy of the dataset:

E (training set) = I(3,6) = 0.9183

b. Information gain of AirTemp, Sky, and Humidity:

E (AirTemp) = 3/9 I(2,1) + 3/9 I(1,2) + 3/9 I(0,3) = 0.6122 → Gain (AirTemp) = E (training set) - E(AirTemp) = 0.3061

E(Sky) = 4/9 I(2,2) + 5/9 I(1,4) = 0.8455 → Gain (Sky) = E (training set) - E(Sky) = 0.0728

E(Humidity) = 6/9 I(3,3) + 3/9 I(0,3) = 0.6667 → Gain (Humidity) = E (training set) - E(Humidity) = 0.2516

Gain (AirTemp) > Gain (Sky) > Gain (Humidity). So we select attribute "AirTemp" with three cases:

- < 20 → N
- 20-30 → ?

Instance	AirTemp	Sky	Humidity	swim
6	20-30	Sunny	Normal	Y
7	20-30	Cloudy	High	N
8	20-30	Cloudy	Normal	N

E(sub set) = I(1,2) = 0.9183
E(Sky) = 1/3 I(1,0) + 2/3 I(0,2) = 0 → Gain(Sky) = E(sub set) – E(Sky) = 0.9183
E(Humidity) = 2/3 I(1,1) + 1/3 I(0,1) = 0.6667 → Gain(Humidity) = E(sub set) – E(Humidity) = 0.2516
Gain(Sky) > Gain(Humidity). So we select attribute "Sky" and continue.

- > 30 → ?

Instance	AirTemp	Sky	Humidity	Swim
1	>30	Sunny	Normal	Y
3	>30	Cloudy	High	N
4	>30	Cloudy	Normal	Y

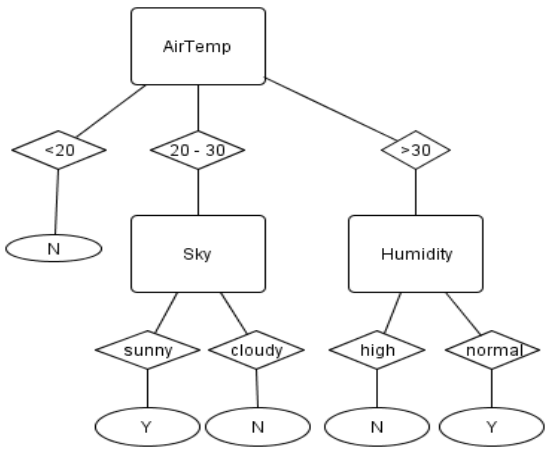
E(sub set) = I(2,1) = 0.9183

E(Sky) = 1/3 I(1,0) + 2/3 I(1,1) = 0.6667 → Gain(Sky) = E(sub set) – E(Sky) = 0.2516

E (Humidity) = 2/3 I(2,0) + 1/3 I(0,1) = 0 → Gain(Humidity) = E(sub set) – E(Humidity) = 0.9183

Gain (Humidity) > Gain (Sky). So we select attribute “Humidy” and continue.

The final decision tree is :



Solution to Question 4: Information Retrieval

Question 4.1 (5pts): Compared to Boolean retrieval:

1. Computationally more expensive [1pt]
2. There is no good way of answering "Does not contain" queries [1.5pts]
3. Boolean conditions cannot be enforced [1.5pts]

Compared to Vector Space retrieval:

1. Computationally more expensive [1pt]

Question 4.2 (10pts):
We find the query vector as $q = (0,1,1,0,1)$. [1pt]
We find $S_2^{-1} = \begin{pmatrix} \frac{1}{2.80} & 0 \\ 0 & \frac{1}{1.94} \end{pmatrix}$. [1pt]
We find $K_2 = \begin{pmatrix} 0.48 & -0.53 \\ 0.46 & -0.21 \\ 0.46 & -0.21 \\ 0.38 & 0.65 \\ 0.45 & 0.46 \end{pmatrix}$. [0.5pt]
We find $D_2^t = \begin{pmatrix} 0.34 & 0.46 & 0.50 & 0.30 & 0.50 & 0.29 \\ -0.38 & 0.46 & -0.49 & 0.22 & -0.15 & 0.57 \end{pmatrix}$. [0.5pt]
We transform the query q in to the two-dimensional concept space: $q^* = q.K_s.S_s^{-1} = (0.4892, 0.0206)$. [2pts]
Now we calculate cosine similarity between q^* and each document in the concept space:

$$sim(q^*, (D_2^t)_1) = 0.6348$$
$$sim(q^*, (D_2^t)_2) = 0.7362$$
$$sim(q^*, (D_2^t)_3) = 0.6841$$
$$sim(q^*, (D_2^t)_4) = 0.8306$$
$$sim(q^*, (D_2^t)_5) = 0.9449$$
$$sim(q^*, (D_2^t)_6) = 0.4905.$$

[3pts]
Therefore, the ranking is $d_5, d_4, d_2, d_3, d_1, d_6$. [2pts]
Question 4.3 (6pts):
First, we find the document vector as $d_7 =$ (Groovy, Ruby, Perl) $= (1,0,0,1,0)$. [1pt]
Second, we transform d_7 into the two-dimensional concept space: $d_7^* = d_7.K_2.S_2^{-1} = (0.3071, 0.0618)$. [2pts]
Now we compute the cosine similarity between q^* and d_7^*:

$$sim(q^*, d_7^*) = 0.9878.$$

[2pts]
Therefore, the ranking is $d_7, d_5, d_4, d_2, d_3, d_1, d_6$. [1pt]
Question 4.4 (4pts):
Yes. [2pts]
1) As new documents are added to the collection, the original SVD might become obsolete as the new documents and their properties and their correlation with the original documents and among themselves is not taken into account. [1pt]
2) If new documents contain new terms, these terms are not included into the original SVD and hence the new terms are omitted. [1pt]