CHALMERS
EXAMINATION / TENTAMEN

Course code/ kurskod	Course name / kursnamn			
EDA387	Computer Networks			
Anonymous code Anonym kod	EDA387-9	Examination date Tentamensdatum	Number of pages Antal blad	Grade Betyg
		2014-01-17	13	3

Solved task Behandlade uppgifter. No / nr		Points per task Poäng på uppgiften.	Observe: Areas with bold contour are to be completed by the teacher. Anmärkning: Rutor inom bred kontur ifylles av lärare.
1	X	9	
2	X	6	
3	X	2	
4	X	2	
5	X	2	
6	X	9	
7	X	8	
8		0	
9	X	6	
10			
11			
12			
13			
14			
15			
16			
17			
18			
Total examination points Summa poäng på tentamen		35	

3

1)a) The user wants to know the name server's host name of the domain chalmers.se 1

1)b) The query is sent to dns.uu.se DNS-server that has IP 130.238.7.10 1

1)c) The anser is Yes because
I) status: NOERROR
II) In answersection there are 4 RRs 1
III) ANSWER: 4

1)d) I) The aa in Flags indicates athoritative answer 1
II) If there is an authoritative answer section in the reply (It appears for example when the query is about MX) Authoritative

1)e) There are 4 RRs
chalmers.se. 172800 IN NS ns1.chalmers.se.
- chalmers.se.: The domain name in question
- 172800: The time in seconds in which this record can be reused. (It is assigned by the administrater of the authoritive DNS-server. In other words how long time in seconds the record will still in the cash of the local DNS-server)
- IN: The type of query (Name server)
- NS: The type of query (Name server)
- ns1.chalmers.se.: The host name of the name server 2

1)f) Time in seconds within which the record can be reused. It is assigned by the administator of the DNS-authoritive server. The purpose is to reduce the load on the server if the RR is cashed in the local DNS server during this time. 1

CHALMERS | Anonymous code / Anonym kod: EDA387-9 | Points for question / Poäng på uppgiften (to be filled in by teacher / ifylles av lärare) | Consecutive page no. / Löpande sid nr: 2 | Question no. / Uppgift nr: 1

1) 9)

RR

NS The query type and answer type is of type (Name server) the name in query is the domain name and the value that is requested is the host name of the domain name server associated with that domain

A The record is of type IPv4 address the name in query is the host name and the value that is requested is the IPv4 address of the host.

AAAA The record is of type IPv6 address The name in query is host name and The value is IPv6 address of the host name

2

CHALMERS | Anonymous code / Anonym kod: EDA387-9 | Points for question (to be filled in by teacher) / Poäng på uppgiften (ifylles av lärare): 6 | Consecutive page no. / Löpande sid nr: 3 | Question no. / Uppgift nr: 2

2) a)

i) 2001:6B0::5E26:AFF:FE66:777C

ii) FF02::1:FF66:777C

iii) FE80::5E26:AFF:FE66:777C

3

2) b)

i) Unicast Global

ii) Multicast link local solicited-node

iii) unicast link local

3

CHALMERS

Anonymous code	Points for question (to be filled in by teacher)	Consecutive page no. Löpande sid nr
Anonym kod EDA387-9	Poäng på uppgiften (ifylles av lärare) 2	Question no. Uppgift nr 3

3.a) The substitution operation is neighbour discovery. It employs two types of ICMPv6 messages, neighbour advertisment and neighbour solicitate messages. A node

Describe 1

3

3.b)

A node has a MAC It sets its interface identifier with help of this mac by putting FFFE between the first 24 and last 24 bits and then inverses the 7th bit from the left. It concatinate it to the prefix that global advertised by the router

1

Describe

messages

link-local

global

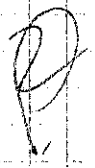
3

4)

6.a) An execution $E = (c(1), a(1), c(2), a(2), \ldots)$, an alternate sequence, such that the step $a(i)$ is applicable to configuration $c(i-1)$ and results in configuration $c(i)$, i.e., $c(i-1) \xrightarrow{a(i)} c(i)$ $(i > 1)$

6.b)

ME. (1) only one node can change it's x variable value in any configuration, and (2) every node can change it's x variable value in infinitly many times in every fair execution in ME.



3

5) a) It is unatractive for the case of internet because its almost impossible or impossible to know the diameter and the number of node.

* In most cases the mode operation is not applied.

5) b) This algorithim is correct because it decreases the number of possible distinct clock values and reduce them to a single value.

If the maximum clock value is less or equal to M-d then the algorithm converges to a safe configuration before the mode application is applied

By using the pigeonhole theorm, in any configuration there must be two clock values x and y such that $y-x \geq d+1$ and there is no other clock values between x and y.

It requires $M-y-1$ steps to convers to a safe configuration.

while do

5)a) Regarding 5. a)

According to the algorithms in the book, It is better to have a bound using the mode operation because a single fault may cause the clock value to reach its maximum value in case we assume the value is unbounded with a finite big variable size. Having a bound and making the clock a constant number of clock values is not a good idea because in some point the clock will not increment but just rotate from left to right. Having a bound that is based on the number of nodes and/or the diameter of the network is proned to not make the algorithm still self-stabilizing.

3

CHALMERS

Anonymous code / Anonym kod: EDA 387-9

Points for question (to be filled in by teacher) / Poäng på uppgiften (ifylles av lärare)

Consecutive page no. / Löpande sid nr: 9

Question no. / Uppgift nr: 6

6) It is self-stabilizing

Answer only one question on this page. Do not write on the back of this paper
Behandla endast en uppgift på detta blad. Skriv ej på baksidan

7)

Lemma 2.4) Les assume by the way of proving by contradiction that for every configuration C in every fair execution that starts in C, P_1 does not change the value of x_1 at least once in every n rounds.

Let C_2 be the configuration that follows immediatly the first time P_2 makes a computation step and changes its x value

It is clear that now that $x_1 = x_2$ in C_2 and all the configuration after C_2 of the $(n-1)$ rounds.

Let C_3 be the configuration that follows immediatly the first time P_3 makes a computation step and changes its x value. Now, it is clear that $x_1 = x_2 = x_3$. This will be repeated until $x_1 = x_2 = x_3 = \cdots = x_n$ which occure in the $(n-1)$th round and presented in configuration C_n.

Now $x_n = x_1$ and because of that P_1 will recompute its x value in the nth rownd which means P_1 changes x_1 in the nth rownd a contradiction

Theorem 2.1)

For every possible configuration C, every fair execution that starts in C reaches a safe configuration with relation to ME within $O(n^2)$ rounds

With accordance to 2.3, For every configuration there exists at least one integer j such that for every P_i $(1 \leq i \leq n)$, $x_i \neq j$.

with accordance to 2.4 For every configuration C, in every fair execution that starts in C, P_1 changes the value of x_1 at least once in every n rounds.

When P_1 increases it's x value (x_1) (when it is equal to x_n) it increments it as $x_1 := (x_1 + 1) \bmod (n+1)$

We must consider every possible value that may be assigned to x_1, and in particular the distinct value j.

Let C_j be the configuration that reached directly after assigning j to x_1.

now it is clear that in all of the configurations that follows C and precedes C_j, and for $2 \leq i \leq n$

For all P_i $x_i = x_{i-1}$

and for $1 \leq i \leq n$ every P_i will copy the value of x_{i-1} and assigns it to x_i, in the configurations that follows C_j.

3

The only possible way that the processors change their values is as follows:

P_2 changes its x_2 value to x_1 $\Rightarrow$ $x_1 = x_2 = j$

then P_3 changes its x_3 value to x_2 $\Rightarrow$ $x_1 = x_2 = x_3 = j$

until configuration n that is reached directly after P_n assigns j to x_n.

C_n is a safe configuration as lemma 2.2

now P_1 will find out that $x_n = x_1$ and it will change its x_1 value to a new distinct value than all other x variables

then the algorithm will reach safe configuration in (n^2+n) rounds

Then the safe configuration is reached with reation to ME within $O(n^2)$ rounds

3

9) a) The sender does not need to get acks observe time out to decide how much to send. It calculates the rate of transmition in byte per second as a function of the round trip time observed, the sigment size, the loss rate as a fraction of lost bytes / sent bytes, and the tcp retransmission rate.

9) b) Let us assume bit torrent it uses out of bound search using a third party. The client download metadata that contains list of peers that has the file or part of it. The client start participating in a peer to peer file-sharing process and start fetch the parts of the file from part of the peers that listed in the metadata. The client needs to upload also to the peers that it download from.

Another example is
A nodeA inform a supernode that it has file X when it joins the network,
When a nother node wants to fetch file X it asks the super node about the place of this file. The super node then informs the asking node that file X can be fetched from node A.