CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Problem	Points/Minutes	Lost	Gained
0	0 (+1)		
1	14 (15 mins)		
2	18 (20 mins)		
3	19 (20 mins)		
4	25 (25 mins)		
5	24 (25 mins)		
Total	100 (+1) (105 mins)		

You may ask for clarification but you are ultimately responsible for the answer you write on the paper. If you make any assumptions state them.

Please look through the entire test before starting. WE MEAN IT!!!

NOTE: M = 10^6 K = 10^3 Mi = 2^{20} Ki = 2^{10}
Illegible answers are wrong answers.

Show your work in the space provided to get any credit for problem-oriented questions.

Good luck!

0. (1 bonus point, 1 min)

Given:
5+3+2 = 151022
9+2+4 = 183652
8+6+3 = 482466
5+4+5 = 202541

What is
8+2+4 = **163246**

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Q1 a,b,c

Cache

1. (14 points, 15 min)

a) (2 points) We have a 4-way set associative cache. How much meta-data is needed per cache line (*aka* set) to implement a TRUE LRU replacement policy? Show your work to get any credit.

4! Choices per cache line => 24 states per line => 5 bits per cache line

+1 if 4! Mentioned
+1 for 5 bits

b) (2 points) For a 32-way set associative cache, we want to implement an approximate LRU policy that protects the datablock that was most recently referenced. Suggest a scheme for doing this. How much meta-data is needed per cache line to implement your scheme? Show your work to get any credit. Your objective is to minimize the amount of meta-data needed per cache line.

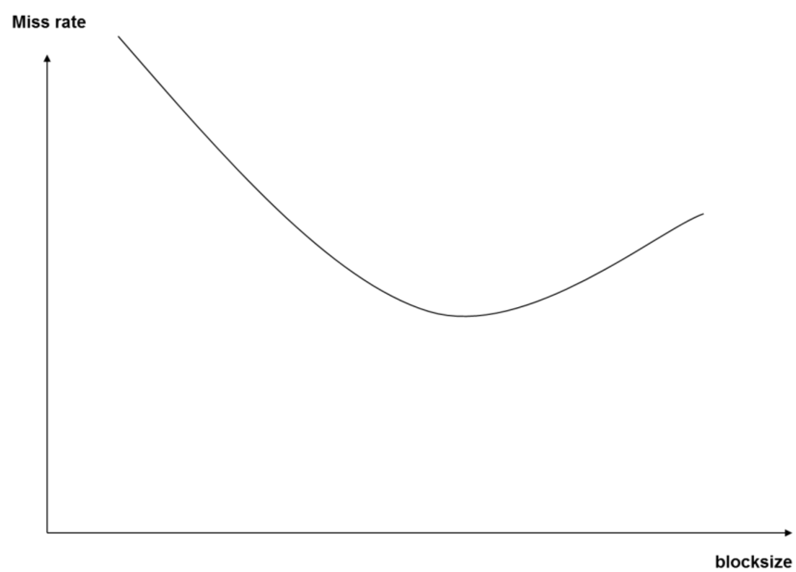
+1 for any solution that works
+1 for minimal (5 bits)

c) As shown in the figure below, for a given cache size increasing the blocksize decreases the miss rate up to a certain point.

(i) (1 point) What is the principle that makes this possible?

Spatial locality

(all or nothing)



(ii) (1 point) In the same figure, beyond a certain point, the miss rate starts increasing beyond a certain blocksize. What is the reason for that phenomenon?

Working set change over the time of execution
(all or nothing)

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Q1 d,e,f,g

d) (3 points) Choose between **conflict miss**, **capacity miss**, and **cold/compulsory miss** for each of the situation described below.

i. There is a cache miss. The cache is full. The missing memory block is being brought for the first time into the cache. This kind of a miss is termed ____**Cold/compulsory**____________________________
ii. There is a cache miss. The cache is NOT full. The missing memory block used to be in the cache before. To find a place to put this word in the cache, an existing datablock has to be evicted. This kind of a miss is termed ___**conflict**_____________________________
iii. There is a cache miss. The cache is full. The missing memory block used to be in the cache before. This kind of a miss is termed _______**capacity**__________________

e) (2 points) Page coloring is a technique for (circle the right choice)

i. Reducing page faults
ii. Increasing the associativity of a virtually indexed physically tagged cache
iii. Increasing the blocksize of a virtually indexed physically tagged cache
iv. Increasing the number of cachelines in a virtually indexed physically tagged cache

f) (1 point) Consider a cache that has 4 memory words in a datablock, and maintains a dirty bit per word. Upon eviction of a datablock from the cache (circle the correct choice)

i. One word is written back to the memory
ii. No word is written back to the memory
iii. 4 words are written back to the memory
iv. The specific words whose dirty bits are set are written back to the memory

g) (2 points) Consider a cache that has 4 memory words in a datablock. The processor has a memory bus that can transfer only ONE word to/from the memory in one memory bus cycle. Upon a cache read miss (circle ALL the correct choices). (+1 for correct choice; -1 for incorrect choice)

i. Only the missing word is brought from the memory into the cache.
ii. The entire memory block containing the missing word is brought into the cache.
iii. The first word brought into the cache is the missing word.
iv. The first word brought into the cache is the first word of the missing memory block.

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Q2 a,b

Input/Output and Disk

2. (18 points, 20 min)

a) (2 points)

In a computer system, the CPU is faster than any I/O device connected to it. Why don't we use programmed I/O to transfer to or from ALL I/O devices? (**Two concise bullets please**)

Processor precious resource (+1)
Data overrun/loss (+1)

Also acceptable answers:
Third party device drivers
Integrating devices connected to I/O bus to the system

b) (6 points)

```
Given the following:
Total number of cylinders in the disk  =  200 (numbered 0 to 199)
        (0 is the outermost cylinder and 199 is the innermost)
Current head position                  =  cylinder 17
        (direction of head motion is towards cylinder 0)
Current requests in order of arrival   =  20, 18, 55, 15, 25, 78, 99
                                          R1, R2, R3, R4, R5, R6, R7
```

(i) (3 points) If the scheduling algorithm is LOOK, what is the expected response time (expressed in number of cylinders traversed) for request R6? (show your work)

Order of request satisfaction:
15, 18, 20, 25, 55, 78, 99

Response time for R6 (at cylinder 78)
= 2+3+2+5+30+23 cylinders traversal time
= 65 cylinders traversal time

-1 for wrong algorithm
-2 for incorrect answer

-1 for arithmetic errors

(ii) (3 points) What is the total head traversal (in terms of number of cylinders traversed) if the scheduling algorithm is SSTF? (show your work)

Head traversal with SSTF for the given reference string:
(Two possibilities)

17 -> 18 -> 20 -> 15 -> 25 -> 55 -> 78 -> 99
⇨ 92 head traversals
17 -> 18 -> 20 -> 25 -> 15 -> 55 -> 78 -> 99
⇨ 102 head traversals

+2 for either sequence
+1 for traversal

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

c) (10 points total) Given the following specification for a disk drive:

```
Average seek time     =  4 ms
Rotational speed      =  7500 RPM
Platters              =  3
Surface per platter   =  2
Tracks per surface    =  8000
Sectors per track     =  1024
Recording density     =  256 bytes per sector
```

Q2 C

(i) (2 points) How much time (expressed in **ms**) is needed to get to a random sector on the disk?

Time to get to a random sector
= Seek time + half rotational latency
= 4 ms + 1/2 * (60*1000)/7500
= (4 + 4) ms
= 8 ms

-1 for not saying 1/2 rot latency

(ii) (2 point) How much time (expressed in **ms**) is needed to read one random sector from the disk when the head is already positioned on the desired sector?

Time to read one random sector once the head is already positioned on the track
= Rotational latency/number of sectors per track
= 8 ms/1024 = 0.0078 ms

-1 for using 1/2 rot. Latency

(iii) (2 points) If the disk gets a request to read 6 random sectors, how much total time (expressed in **ms**) will that request take to complete?

Time to read 6 random sectors
= 6 * (time to get to a random sector + sector read for time)
= 6 * (8+.0078) ms
= 48.047 ms

+1 for right formula; +1 for correct answer

(iv) (2 points) If the disk gets a request to read 6 consecutive sectors, how much total time (expressed in **ms**) will that request take to complete?

Time to read 6 contiguous sectors
= Time to get to a random sector on the disk
+ time to read 6 contiguous sectors
= 8 + 6 * 0.0078
= 8.047 ms

+1 for right formula; +1 for correct answer

(v) (2 points) What is the transfer rate of the disk?

Transfer rate
= Bytes per track/rot latency
= (1024 * 256)/8 bytes/ms
= 1000 * (1024 * 256 / 8) bytes/s
= 32768000 bytes/s

+1 for right formula; +1 for correct answer

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

File Systems (see last page for a cheat sheet on Unix FS commands)

3. (19 points, 20 min)

a) (1 point)

Q3 a, b

The current state of a file "foo" is as follows:

```
    ---xr-x---   3 rama            0 Apr 27 21:01 foo
```

After executing the following commands:

```
chmod w+w foo
chmod u+w foo
```

The new access rights are (circle the right choice):

(i) --wxrwx-w-

(ii) -rwxr-x---

(iii) --wxr-x-w-

(iv) ---xr-x-w-

b) (8 points) In the following table, assume **none of the files exist to start with** in the current directory. Fill in the table. The reference count in the table pertains to the i-node that is affected by the command in that row. If a new i-node is created, show the old reference count for that i-node as 0.

COMMAND	NEW I-NODE CREATED (YES/NO)	REFERENCE COUNT OLD	NEW
touch f1	Yes	-	1
touch f6	Yes	-	1
ln f1 f2	No	1	2
ln -s f1 f3	Yes	-	1
ln -s f3 f4	Yes	-	1
ln f6 f5	No	1	2
rm f6	i-node f6 deleted	-	-
rm f2	i-node f1/f2 deleted	-	-

Use this area for rough work for this question

Grading rubric:

touch command => create i-node +1; ref count +1

ln -s => create i-node +1; ref count +1

ln => no i-node created +1; ref count inc +1

rm => delete an i-node +1 (for each rm)

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

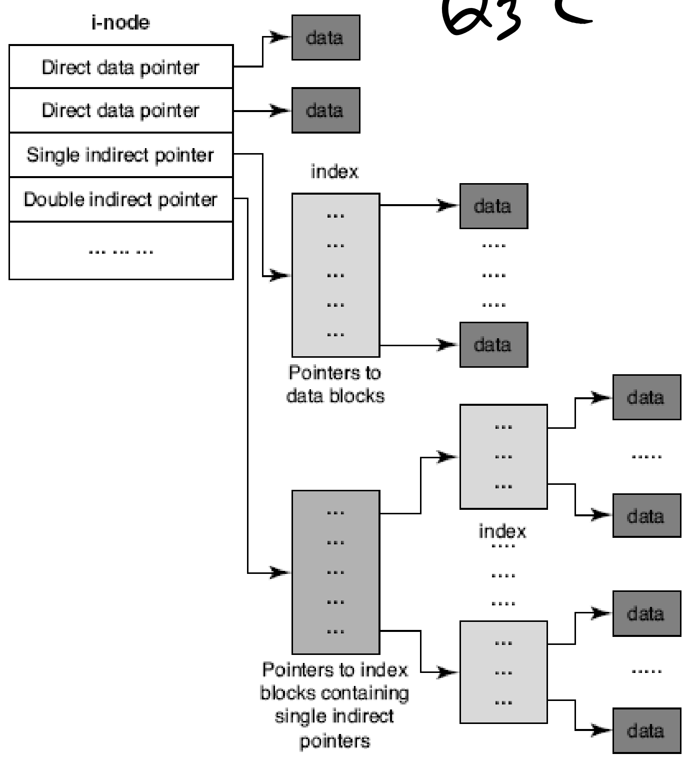
Name:_____TAs & Kishore___GT Number: _______________

Q3 C

c) (**NOTE Ki = 2^{10} Mi = 2^{20}**)

```
Given the following:
  Size of index block = 256  bytes
  Size of Data block  = 4096 bytes
  Size of pointer     = 8 bytes (to
                index or data blocks)
  The i-node consists of
      2 direct data block pointers,
      1 single indirect pointer, and
      1 double indirect pointer.
```

Note that the index blocks and data blocks are allocated on a need basis. An index block is used for the top-level i-node as well as for the index blocks that store pointers to other index blocks and data blocks (see Figure).



i. (2 points) How many data blocks are used to store a 100 KiB file?

Number of data blocks
= 100 * 1024/4096
= 25 data blocks

+1 for formula
+1 for answer

-1 for using K instead of Ki

ii. (2 points) How many index blocks (including the i-node for the file) are needed to store a 1 MiB file?

<u>Index block calculation</u>

Start with one index block for the file's i-node
(Number of index block = 1)
Number of data blocks needed for 1 MiB file
= 1 * 2^{20}/4096
= 256 data blocks
Number of non-direct data blocks needed
= 256 - 2
= 254

Q3 c (Contd.)

Add 1 first level index block

(Number of index block+1, now at 2)

With one first level index block we can get 32 data blocks

Remaining data blocks to be gotten by adding a second level index npde
Number of data blocks to be gotten by adding a second level index node
= 254-32
= 222
Add 1 second level index block

(Number of index block+1, now at 3)

Number of first level index blocks needed to accommodate 222 data blocks
= ceil (222/32)
= 7

(Number of index block+7, now at 10)

Total number of index blocks needed = 10

+1 for methodology
+1 for correct answer

iii. (2 points) What is the largest file size that can be supported in this file system?

Largest file
= 2 direct data blocks + 32 data blocks from first level
+ 32 * 32 data blocks from second level
= 2+32+1024
= 1058 data blocks
= 1058 * 4096 bytes

+1 for methodology
+1 for correct answ

d) (4 points) Strictly Contiguous allocation of disk blocks (using a best-fit allocation policy depending on expected file size) results in (circle ALL the TRUE statements in the following list) (+1 for correct choice; -1 for incorrect choice)

Q3 d

(i) External Fragmentation
(ii) Internal fragmentation
(iii) Ability to grow the file easily
(iv) Minimal time overhead for allocation
(v) Good sequential access to the files
(vi) Good random access to the files
(vii) Simple data structures for file maintenance
(viii) Very good space efficiency

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Q4 a, b, c

Parallel Systems

4. (25 points, 25 mins)

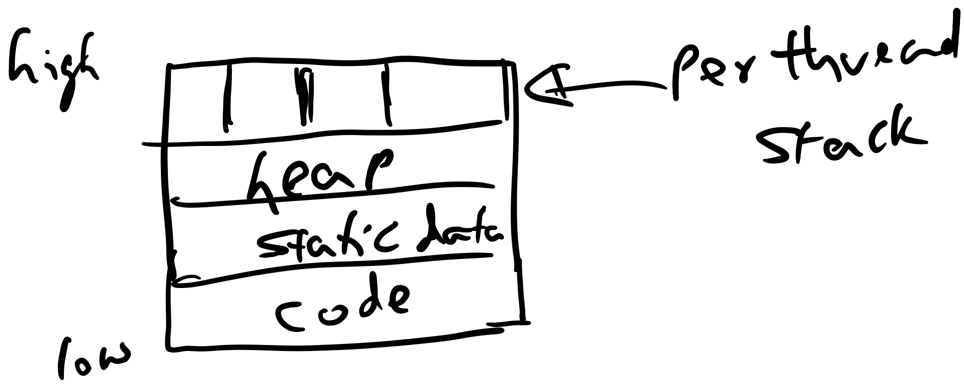
a) (3 points)

(fill in selecting the appropriate phrases from these choices: **threads, process, processes, thread, memory footprint, VPN, PFN**

The state of a multi-threaded program is fully defined by its memory footprint and the state of the threads executing within the process.

b) (2 points)

You click on the icon of a multi-threaded program.
A program is currently in execution with 4 threads in it.
The OS creates a memory footprint of the program reading it from the disk.
Pictorially show this memory footprint.



+2 for showing per thread stack

c) (4 points)

Given the code fragment:

```
acquire_shared_resource()
{
  thread_mutex_lock(cs_mutex);
    if (res_state == BUSY)
      thread_cond_wait (res_not_busy, cs_mutex);
    res_state = BUSY;
  thread_mutex_unlock(cs_mutex);
}
```

A thread wakes up from the cond_wait shown in the code above.

(i) What are the invariants expected by this thread?

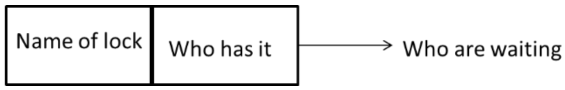
- **It has cs_mutex** **(+1)**
- **res_state is not equal to BUSY** **(+1)**

(ii) Who ensures these invariants are satisfied?

- **OS is responsible ensuring that the thread has cs_mutex** **(+1)**
- **The program developer is responsible for ensuring that the res_state is changed to not BUSY before signaling the thread** **(+1)**

Q4 d, e

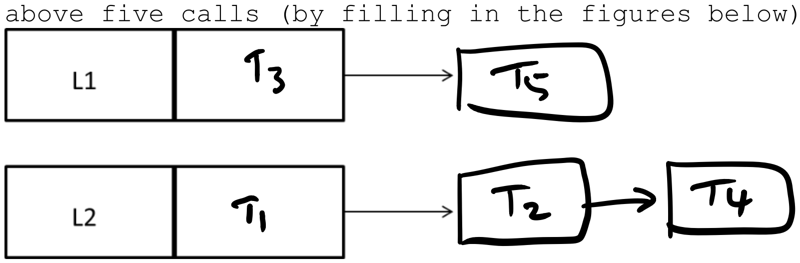
d) (5 points) The internal representation in the thread library for a LOCK is shown below:



Assume that the following events happen in the order shown (T1-T5 are threads of the same process):

- T1 executes thread_mutex_lock(L2);
- T2 executes thread_mutex_lock(L2);
- T3 executes thread_mutex_lock(L1);
- T4 executes thread_mutex_lock(L2);
- T5 executes thread_mutex_lock(L1);

Assuming that there have been no other calls to the threads library prior to this, show the state of the internal data structures of L1 and L2 after the above five calls (by filling in the figures below).



-1 for each incorrect entry

e) (6 points)
A multithreaded process is executing on an SMP which supports hardware cache coherence and per processor TLB. Each thread is on a distinct processor.
(i) (1.5 points) All threads have to execute within the same address space. How is this ensured?

The OS maintains a single page table in shared memory for the process. Since the page table is in shared memory it is visible to all processors. Thus all threads will have a consistent identical view of the shared address space.

-1 if single page table in shared memory for the process is not mentioned

(ii) (1.5 points) All the threads have to see the same mapping from VPN to PFN regardless of which processor they are executing on. How is this ensured?

The TLB is private to each processor. Upon a translation miss, the missing PTE (page table entry) is brought into the TLB. Upon a page fault, if the OS on one of the processors evicts a page to make room for the missing page, then the PTE will change for the evicted page to invalid. If that entry is in the TLB of some other processor then it is important to invalidate that entry. Since the TLB is private to each processor, the OS (page fault handler that made the change to the page table) has to do the needful by running a "TLB shootdown" algorithm that informs its peers on the other processors to invalidate the old VPN->PFN mapping if it exists on that processor.

-1 if OS responsible for keeping the TLBs consistent

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Q4 e (Contd.)

(iii) (1.5 points) The OS implements mutual exclusion lock using a Test-and-Set instruction available in the processor architecture. The threads have to have synchronization atomicity for a given lock variable regardless of which processor they are executing on. How is this ensured?

Implementation of T&S instruction by-passes the cache and directly uses the value in memory. Since all processors have to use the shared bus (one at a time) to get to shared memory, by-passing the cache ensures atomicity for T&S instruction emanating from multiple processors.

-1 if by-passing the cache for T&S not mentioned

(iv) (1.5 points) All the threads have to see the same value in a given memory location. How is this ensured?

In an SMP, the hardware implements cache coherence protocol using either an invalidation-based or update-based protocol. Thus all processors will see the same value in a given memory location.

-1 if hardware cache coherence not mentioned

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Q4 f

f) (5 points)

Given the following details about an SMP (symmetric multiprocessor):
Cache coherence protocol: write-invalidate
Cache to memory policy: write-back
Initially:
The caches are empty
Memory locations:
A contains 20
B contains 18

Consider the following timeline of memory accesses from processors P1, P2, and P3.

Time (in increasing order)	Processor P1	Processor P2	Processor P3
T1	Load A		
T2		Load B	
T3			Load A
T4	Store #80, A		
T5		Store #30, B	

Using the table below, summarize the activities and the values in the caches and memory.

Time	Variables	Cache of P1	Cache of P2	Cache of P3	Memory	
T1	A	20	NP	NP	20	+1
	B	NP	NP	NP	18	
T2	A	20	NP	NP	20	
	B	NP	18	NP	18	+1
T3	A	20	NP	20	20	+1
	B	NP	18	NP	18	
T4	A	80	NP	I	20	+0.5 each
	B	NP	18	NP	18	
T5	A	80	NP	I	20	
	B	NP	30	NP	18	+1

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Q5 a

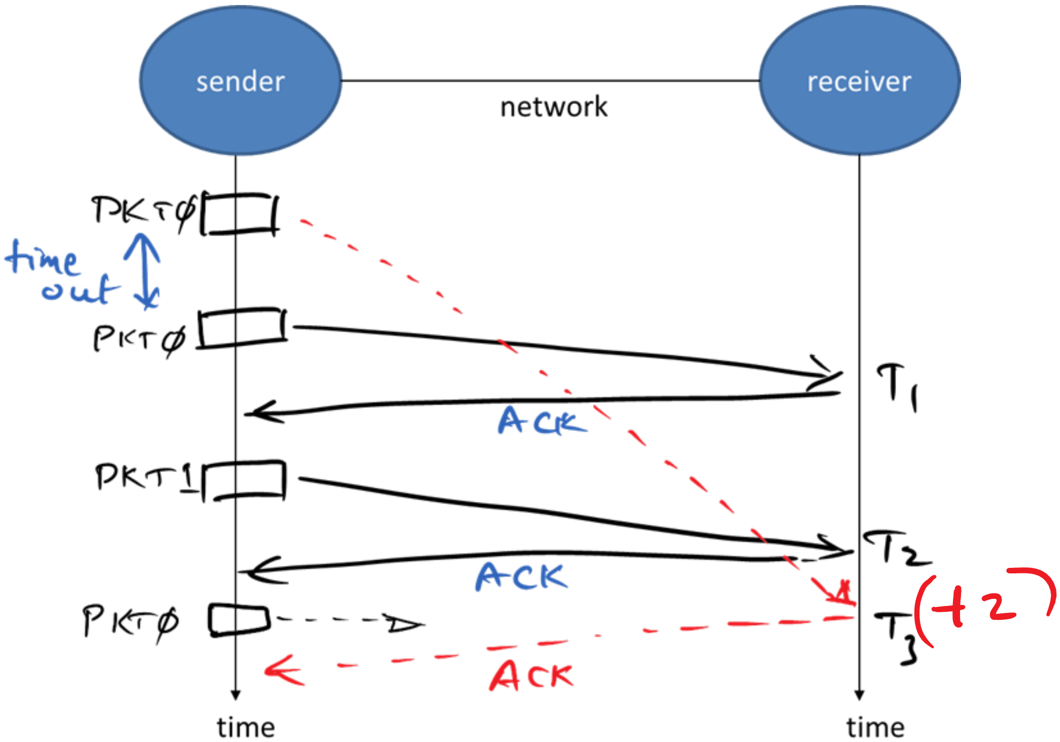
Networking

5. (24 points, 25 min)

a) (3 points) The one-bit sequence number requirement for the "Stop-and-wait protocol (aka alternating bit protocol)" is predicated on the following assumption:

- packets do not get reordered in the network OR arbitrarily delayed

Show using the space-time diagram below what can go wrong if this assumption does not hold.



- **Sender sends PKT0**
- **Sender times out; retransmits PKT0; Receiver ACKS PKT0**
- **Receiver is now awaiting sequence number 1**
- **Sender sends PKT1; Receiver ACKS PKT1**
- **Receiver is now awaiting sequence number 0**
- **The original PKT0 arrives LATE (time T3) at receiver**
- **Receiver accepts it as a new packet and ACKs it**
- **Obviously this is in error**

(+1 grace point) (+2 for showing anomaly like above)

Q5 b, c, d, e

b) (2 points) What determines the timeout for packet retransmission in the transport protocol? (**one concise bullet please**)

Expected latency of transmission to the desired destination.

(all or nothing)

c) (5 points) Match the left side to the right side with respect to transport protocols (**draw an arrow from the concern on the left side to the correct solution on the right side**)

(+1 for each correct arrow)

Concerns or Issues	**Solution**
Arbitrary message size	Sequence numbers
Out of order delivery	Checksum
Packet loss	Windowing mechanism
Bit errors in transmission	Multiple packets
Queueing delays	Positive ACKs and Buffering

d) (2 points) (**Answer True/False with justification**)
With reference to the packet header generated by the transport layer shown below

```
struct header_t {
  int destination_address;
  int source_address;
  int num_packets;
  int sequence_number;
  int packet_size;
  int checksum;
};
```

It is necessary to have the field "num_packets" in the header only in the first packet.

False. Since packets may arrive out of order, the destination needs to know how many packets are in a multi-packet message regardless which packet arrives first.

+1 for False
+1 for reason

e) (2 points) Why do we need MAC address when each node on the Internet has an IP address?

IP address is a virtual address used by the network layer of the protocol stack; **(+1)**
MAC address is the real address used by the NIC (which is at the link layer) to receive a packet on the wire intended for this node. **(+1)**

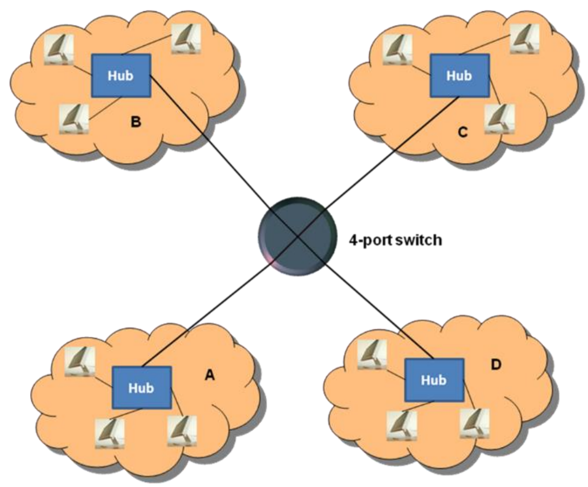
Q5 f, g, h, i

f) (2 points) The load on a LAN connecting computers is very light. Which link layer protocol (Ethernet or Token ring) would you recommend and why?

Ethernet. **(+1)**
Results in lower latency at low network loads. **(+1)**

g) (1 point) With respect to this figure

The number of collision domains (circle the right choice):



(i) One

(ii) Two

(iii) Three

(iv) Four

(v) Twelve

h) (2 points) (**Answer True/False with justification**) Assume computers are connected by Ethernet as the link layer protocol. TCP/IP is used for them to communicate on the Internet. Since Ethernet does collision detection there is no need to check for any packet corruption in the Transport layer.

False. **(+1)**

Collision detection by Ethernet is only within the given collision domain.

Packet corruption could happen due to bit errors introduced in transmission on wide are Internet.

(+1) if either of the above two reasons given.

i) (1 point) In IPv4, an IP address is given by the notation:
a.b.c.d

The number of bits to represent this IP address is (**circle the right choice**):

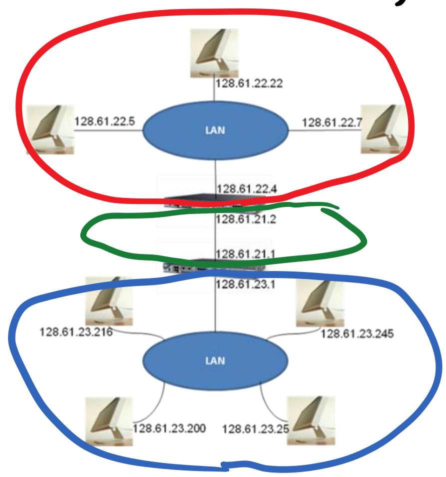
(i) 64 bits
(ii) 32 bits
(iii) 128 bits
(iv) 256 bits

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: ____________ Q5 j,k

j) (2 points) Given the figure:

Identify the distinct IP networks in the figure by **drawing an oval** around each



k) (2 points)
Given the following:

Message size	=	20,000 bits
Bandwidth on the wire	=	200,000 bits/sec
Time of flight	=	10 msecs
Sender overhead	=	1 ms
Receiver overhead	=	1 ms

Assuming no packet loss, compute the throughput.

Time to send one packet
= sender overhead + wire delay + time of flight + receiver overhead
= 1 ms + 20000 * 1000/200000 ms + 10 ms + 1 ms
= 1 ms + 100 ms + 10 ms + 1 ms
= 112 ms

+1 for formula
+1 for plugging in the right numbers (even if not simplified)

CS 2200 Spring 2017 Final (B) 8 AM to 10 AM

Name:_____TAs & Kishore___GT Number: _______________

Unix command	Semantics	Elaboration
touch <name>	Create a file with the name <name>	Creates a zero byte file with the name <name> and a creation time equal to the current wall clock time
mkdir <sub-dir>	Create a sub-directory <sub-dir>	The user must have write privilege to the current working directory (if <sub-dir> is a relative name) to be able to successfully execute this command
rm <name>	Remove (or delete) the file named <name>	Only the owner of the file (and/or superuser) can delete a file
rmdir <sub-dir>	Remove (or delete) the sub-directory named <sub-dir>	Only the owner of the <sub-dir> (and/or the superuse) can remove the named sub-directory
ln –s <orig> <new>	Create a name <new> and make it symbolically equivalent to the file <orig>	This is name equivalence only; so if the file <orig> is deleted, the storage associated with <orig> is reclaimed, and hence <new> will be a dangling reference to a non-existent file
ln <orig> <new>	Create a name <new> and make it physically equivalent to the file <orig>	Even if the file <orig> is deleted, the physical file remains accessible via the name <new>
chmod <rights> <name>	Change the access rights for the file <name> as specified in the mask <rights>	Only the owner of the file (and/or the superuser) can change the access rights
chown <user> <name>	Change the owner of the file <name> to be <user>	Only superuser can change the ownership of a file
chgrp <group> <name>	Change the group associated with the file <name> to be <group>	Only the owner of the file (and/or the superuser) can change the group associated with a file
cp <orig> <new>	Create a new file <new> that is a copy of the file <orig>	The copy is created in the same directory if <new> is a file name; if <new> is a directory name, then a copy with the same name <orig> is created in the directory <new>
mv <orig> <new>	Renames the file <orig> with the name <new>	Renaming happens in the same directory if <new> is a file name; if <new> is a directory name, then the file <orig> is moved into the directory <new> preserving its name <orig>
cat/more/less <name>	View the file contents	