

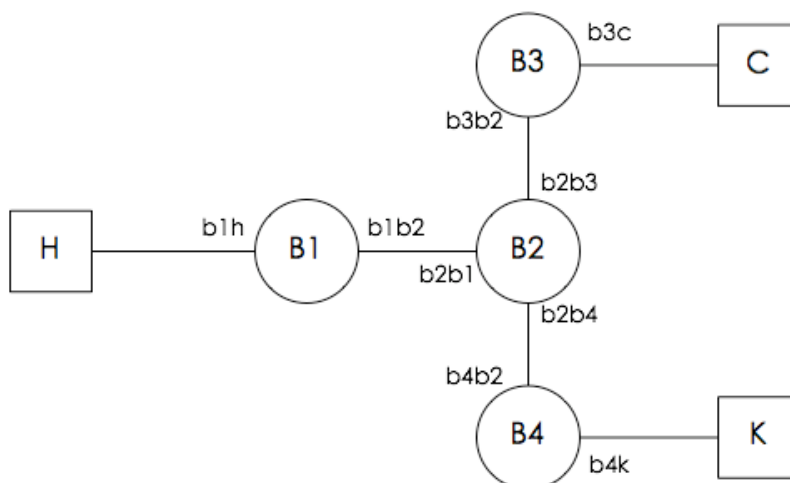
姓名: wissen@FreeEE 班级: Pretest 学号: N/A.

## 《计算机网络》2005年期末试题及参考答案

以下已经预备好表格的题目请直接在试卷上填表作答，  
其余问题则在答题纸上回答，试卷必须同答题纸一并交回。

(2005年12月30日 19:00 – 21:15)

- 一. H和C之间的网络包含若干个网桥（交换机），同时还连接有主机K，其拓扑结构如图，其中每个网桥的端口不妨用它自己和它邻近的节点标记。回答后面的问题。（25分）



1. 假设初始状态下所有网桥上的散列表都是空的，然后依次发生以下事件。其中每一步传送过程中，每个网桥对相应数据帧的处理是扩散还是转发？这一步完成后以后B1 ~ B4各网桥上的散列表情况如何？请填写每一步后面的表格作答（表格中给出了一个例子）。（15分）：

- a. K发送报文给C；

网桥		B1	B2	B3	B4
对该帧 的处理	接收与否	接收	接收	接收	接收
	扩散或转发	扩散	扩散	扩散	扩散
处理完 成后的 散列表	主机/端口	K/b1b2	K/b2b4	K/b3b2	K/b4k
	主机/端口	—	—	—	—
	主机/端口	—	—	—	—

b. C发送报文给K;

网桥		B1	B2	B3	B4
对该帧的处理	接收与否	收不到	接收	接收	接收
	扩散或转发	—	转发	转发	转发
处理完成后的散列表	主机/端口	K/b1b2	K/b2b4	K/b3b2	K/b4k
	主机/端口	—	C/b2b3	C/b3c	C/b4b2
	主机/端口	—	—	—	—

c. H发送报文给C。

网桥		B1	B2	B3	B4
对该帧的处理	接收与否	接收	接收	接收	收不到
	扩散或转发	扩散	转发	转发	—
处理完成后的散列表	主机/端口	K/b1b2	K/b2b4	K/b3b2	K/b4k
	主机/端口	H/b1h	C/b2b3	C/b3c	C/b4b2
	主机/端口	—	H/b2b1	H/b3b2	—

**考查点：后向学习法维护链路层的散列表。**

**备注：“收不到”可以写“不接收”或者“否”，但这时候不能说对该帧“丢弃”，因为无从丢弃**

- 假定上述网络是10Mbps的以太网。以太网的争用时隙是  $2\tau = 51.2\mu s$ ，从而意味着最小的帧长是64字节。快速以太网保持最小帧长不变，却使速率提高到100Mbps，是如何实现的？代价是什么？

**提高时钟频率，争用时隙也减少到原来的1/10。代价是线路传输允许的最大距离也相应降低。**

**考查点：争用时隙的概念**

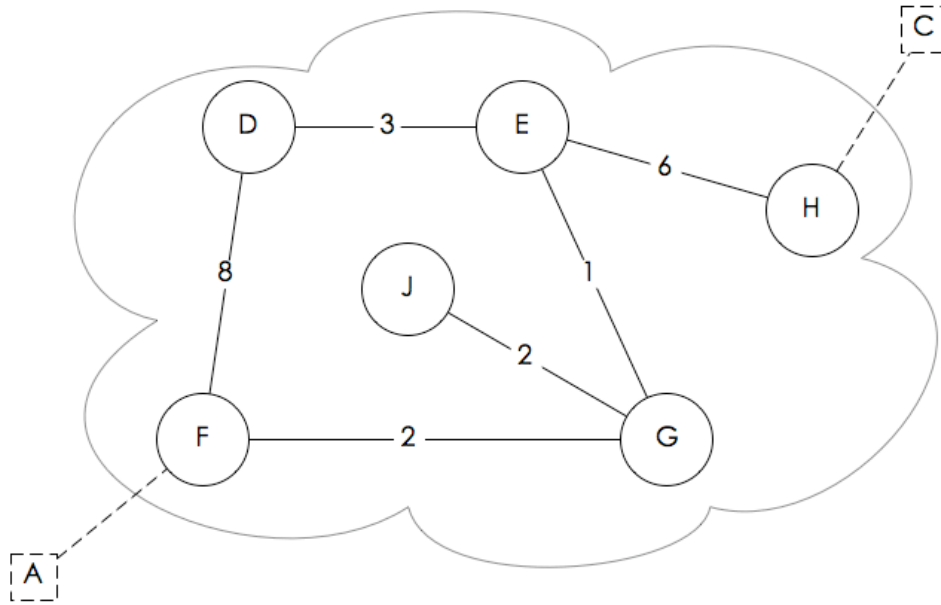
- 如果保持最大传输距离不变，而提高信号的时钟频率，使之达到100Mbps，则最小帧长将如何变化？有何不妥？

**传输距离决定可以检测到冲突的最小时间，因此，保持传输距离不变提高速率到原来的10倍，要求最小帧长增大到640byte，这样可能会导致传输失败的代价提高，并且由于包长比帧长小填充后发送浪费带宽，导致链路的信道利用率降低；此外，最小帧长的变化，还会引起和10M以太网的不兼容问题。**

考查点：帧长和争用时隙的关系

备注：不兼容问题不做要求

二. 某网络N的细节结构如下图云中的部分，主机A和C通过这个网络可以实现互连，链路上的代价已经标注出来，试回答后面的问题。（20分）



1. 假定节点D ~ H之间使用链路状态协议，试写出节点 F 收到所有其它节点的链路状态分组（LSP）之后进行前向搜索计算的全过程，即从初始化开始，每一步读取谁的LSP，如何更新已确认表M和临时表T，以下表格给出了最初的情况，其中M和T都是（目标，代价，下一跳）这样的三元组的集合，把这个表格的演算继续下去直到计算结束（表格中已经给出了初始化的情况，计算结束的位置上请标记“End”）。

步骤	M	T
初始化	(F,0,-)	∅
1、遍历F的LSP	(F,0,-)	(D,8,D), (G,2, G)
2、取最小代价	(F,0,-), (G,2, G)	(D,8,D)
3、遍历G的LSP	(F,0,-), (G,2, G)	(D,8,D),(J,4,G),(E,3,G)
4、取最小代价	(F,0,-), (G,2, G), (E,3,G)	(D,8,D),(J,4,G)
5、遍历E的LSP	(F,0,-), (G,2, G), (E,3,G)	(D,6,G),(J,4,G),(H,9,G)
6、取最小代价	(F,0,-), (G,2, G), (E,3,G), (J,4,G)	(D,6,G), (H,9,G)
7、遍历J的LSP	(F,0,-), (G,2, G), (E,3,G), (J,4,G)	(D,6,G), (H,9,G)

8、取最小代价	(F,0,-), (G,2, G), (E,3,G), (J,4,G), (D,6,G)	(H,9,G)
9、遍历D的LSP	(F,0,-), (G,2, G), (E,3,G), (J,4,G), (D,6,G)	(H,9,G)
10、取最小代价	(F,0,-), (G,2, G), (E,3,G), (J,4,G), (D,6,G), (H,9,G)	
	End	

考查点：LS路由及前向搜索算法

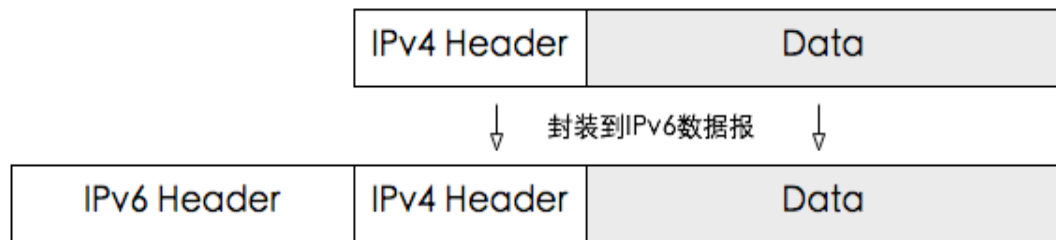
备注：步骤列的说明不做要求

2. 假定节点之间使用距离向量协议，而该协议与RIP不同，它考虑链路上的代价，而不简单以跳数作为计算依据，写出节点 F 从初始状态开始到收敛为止，其距离向量路由表的变化过程，下面的表格给出了初始的情况，用“代价/下一跳”的形式把它填写完整（达到收敛后请在该行任一列标注“收敛”）。

目的地 时刻	D	E	G	H	J
初始化	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
1个周期后	8/D	$\infty$	2/G	$\infty$	$\infty$
2个周期后	8/D	3/G	2/G	$\infty$	4/G
3个周期后	6/G	3/G	2/G	9/G	4/G
4个周期后	收敛				
5个周期后					
6个周期后					

考查点：DV路由

- 三. TCP/IP的体系结构允许IP运行在任何协议上，包括其本身或者IPv6。假定中间的网络N完全运行IPv6，由于主机A和C之间还通过IPv4相互通信，可以把IPv4整个协议数据单元**不做任何改变**封装到IPv6的分组中去（如下图）。现在知道A通过F把数据报封装成IPv6并送入网络，又通过H解开封装恢复成IPv4的数据报，IPv4数据报里的TTL在经过整个网络N的时候都不会发生变化，直到从H离开时才会减 1，所以从IPv4的角度来看，F和H之间只有一跳，相当于一段虚拟的点到点链路。考查后面的问题。（10分）



1. IPv6的地址和路由遵循CIDR的体系结构。已知节点F上通过手工操作定义了如下的转发表，对于目标地址2001:da8:f001:f003::233，下一跳应送往何处？简要说明你的计算过程或者理由。

目标地址	下一跳
3ffe:3200::/32	G
3ffe:3200:8000::/36	D
2001:250::/32	D
2001:c00::/23	D
2001:e00::/23	G
2000::/3	G

**D。依据最长匹配的转发原则，该目的地址匹配到 2001:c00::/23 条目。**

**考查点：CIDR地址结构；最长匹配转发**

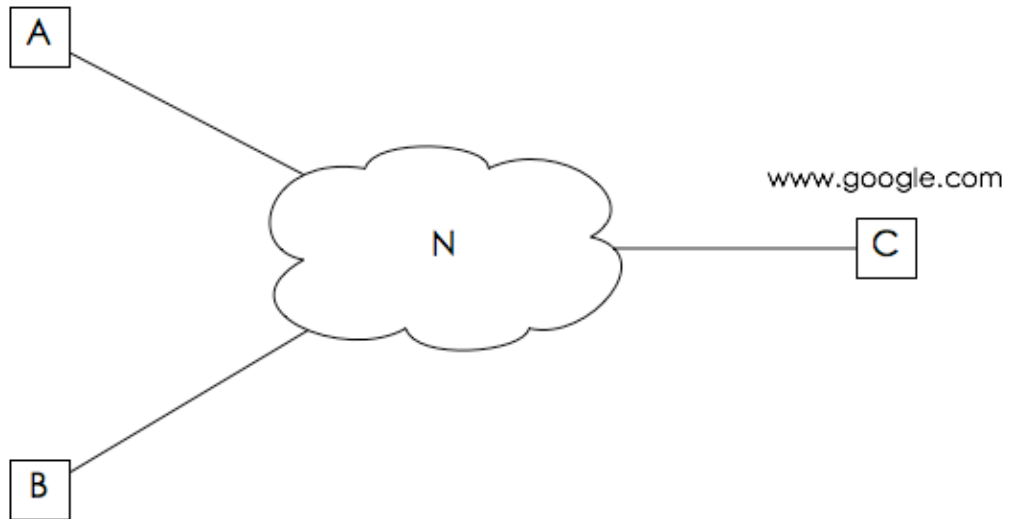
2. 简单说明：这种封装对于从A到C的TCP最大数据段长度（MSS）有何影响？此外，如果整个IPv6网络N全部由以太网实现，F和H之间的虚拟IPv4链路上MTU是多少？

**由于IPv6封装导致端到端的MSS减小IPv6基本头长度，即40字节。如果N全部由Ethernet实现，则F和H之间的虚拟IPv4链路上的MTU应扣除IPv6的基本头长度，为1460字节。**

**考查点：封装；Overlay的思想**

- 四. 如下图，主机A将要访问主机C上的主页http://www.google.com/，已知主机A已经将主机B设置为给自己提供域名解析的服务器。在这个访问过程中，在主机A或B上通过tcpdump工具记录到若干与此有关的分组，其中部分分组的摘要（所谓摘要，是分组从IP头开始的一定长度以内的片断）如下，每个摘要已经注明其发送源（src）和目的地（dst）。根据这些摘要，结合Internet应用及DNS服务的一般过程，以及有关协议头的知识，回答下列问题。（试卷最后附有常用的协议号和端口号的十进制值列表，可供参考。）（20分）：

A's name server = B



(1). src: A; dst: B

```
4500 003c 14b5 4000 4011 86eb a66f cb68
ca26 6312 800f 0035 0028 9f4a 8181 0100
0001 0000 0000 0000 0377 7777 0667 6f6f
676c 6503 636f 6d00 0001 0001
```

(2). src: A; dst: C

```
4500 003c db4c 4000 4006 513b a66f cb68
42f9 5963 80dd 0050 8571 c621 0000 0000
a002 16d0 4bd0 0000 0204 05b4 0402 080a
ed02 1d6d 0000 0000 0103 0302
```

(3). src: A; dst: C

```
4500 0028 db52 4000 4006 5149 a66f cb68
42f9 5963 80dd 0050 8571 c750 727c 3487
5010 1b3c 1171 0000
```

(4). src: A; dst: C

```
4500 0156 db50 4000 4006 501d a66f cb68
42f9 5963 80dd 0050 8571 c622 727c 3241
5018 16d0 0f7d 0000 4745 5420 2f20 4854
5450 2f31 2e30 0d0a 486f 7374 3a20 7777
772e 676f 6f67 6c65 2e63 6f6d 0d0a 4163
6365
```

(5). src: A; dst: C

```
4500 0028 db4e 4000 4006 514d a66f cb68
42f9 5963 80dd 0050 8571 c622 727c 3241
5010 16d0 1951 0000
```

```
(6). src: B; dst: A
4500 00d0 24b5 0000 3811 be57 ca26 6312
a66f cb68 0035 800f 00bc 94ff 55a8 8180
0001 0003 0006 0000 0377 7777 0667 6f6f
676c 6503 636f 6d00 0001 0001 c00c 0005
0001 0000 0381 0008 0377 7777 016c c010
c02c
```

```
(7). src: C; dst: A
4510 026d a0a1 0000 3506 d4a5 42f9 5963
a66f cb68 0050 80dd 727c 3241 8571 c750
5019 1920 12ef 0000 4854 5450 2f31 2e30
2033 3032 2046 6f75 6e64 0d0a 4c6f 6361
7469 6f6e 3a20 6874 7470 3a2f 2f77 7777
2e67
```

```
(8). src: C; dst: A
4500 002c dd82 0000 f406 db14 42f9 5963
a66f cb68 0050 80dd 727c 3240 8571 c622
6012 1ffe f865 0000 0204 05b4 0000
```

```
(9). src: C; dst: A
4500 0028 d590 0000 f406 e30a 42f9 5963
a66f cb68 0050 80dd 727c 3241 8571 c750
5010 1ed0 1023 0000 0000 0000 0000
```

1. 根据上述分组摘要，试给出A，B，C三个主机的IP地址，用点分十进制的方式表示。

A: 166.111.203.104

B: 202.38.99.18

C: 66.249.89.99

**考查点：识别真实分组中的协议头的能力；IPv4地址的表示**

2. 上述分组摘要分别表示了网络上发生的与此访问有关的事件，但遗憾的是这些分组不是按事件发生的顺序摆放的，请对它们按照发生的时间先后进行排序。写出求解过程或理由。

**参考解答：**

i. 首先根据访问应用服务先访问DNS，确定1, 6 在先；

ii. 根据A和C发出的序列号，可知：

1. A发出的：2, {4, 5}, 3

2. C发出的：8, {7, 9}；

(其中,  $\{x, y\}$  表示两个元素的一个置换, 即  $x, y$  或者  $y, x$ )

iii. 然后根据TCP的标志位等信息确定正确的顺序:

1. 注意到 (2) 是 SYN, ! ACK, 而 (8) 是 SYN, ACK, (8) 是对 (2) 的应答;
2. 观察 (8) 的 Acknowledgement, 可知 (4) 和 (5) 是紧接 (8) 之后的分组, 其中注意到 (4) 的数据长度大于 0, 因此 (4) 如果在 (5) 之前二者就不可能有相同的序列号, 所以 (5) 在 (4) 之前;
3. 注意到 (4) 的 PSH, 它可能在 (7) 和 (9) 之前, 这一点可以通过 (7)、(9) 的序列号恰是 (4) 所期望的来确定 (或者通过 (7)、(9) 的 Acknowledgement 比 (4) 的序列号大得多来确定);
4. 再看 (3), 注意到它恰好具有 (7)、(9) 都所期待的序列号, 所以它在 (7)、(9) 的后面;
5. (7) 和 (9) 的顺序仍然用判断 (4) 和 (5) 之间的情况来考虑, 可知 (9) 在 (7) 之前。

所以, 正确的顺序是 1, 6, 2, 8, 5, 4, 9, 7, 3

考查点: 应用服务的一般过程及其与DNS的关系; TCP的时序; 序列号和应答号的关系; TCP的标志位; TCP/IP实践中的简单推理能力

备注: 通过 (7) 具有 FIN 标志而判断 (9) 在前的逻辑是不成立的, 因为发出 FIN 以后完全还可能发出不带数据的对对方的数据分组或者 FIN 分组的确认。

3. A访问C上的主页, 从应用层看是面向连接还是无连接的?

HTTP是无连接的应用层协议 (尽管它以面向连接的TCP为基础)。

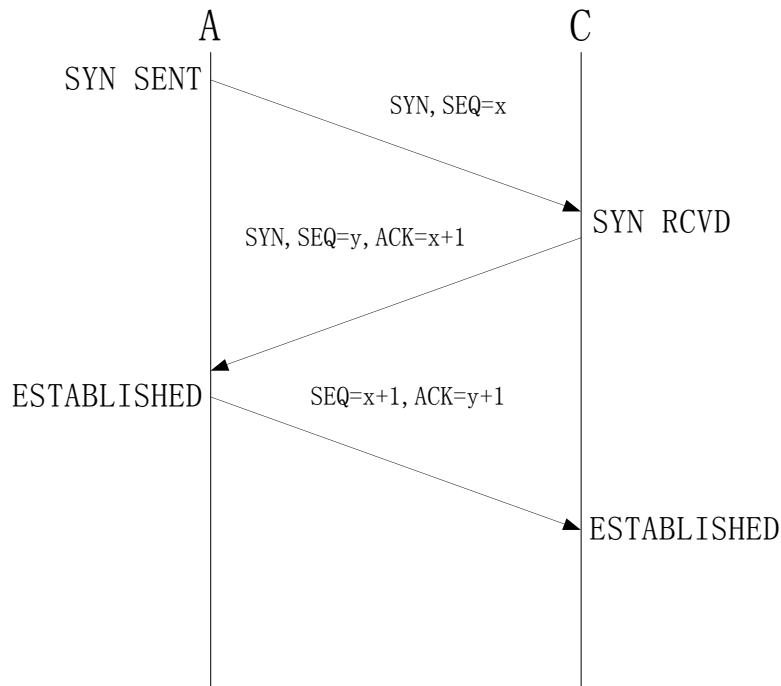
考查点: 体系结构不同层 (layer) 上, 面向连接/无连接的意义不可混淆

五. 根据有关TCP的知识, 回答下列问题 (25分):

1. TCP建立连接采用的是什么方式? 画出两个主机X, Y之间由X首先发起建立一个TCP连接的一般时序图, 并在图的两侧注明不同时间上主机X和Y分别处于什么状态。

TCP连接采用三次握手方式。时序图:





2. 上一题第 (7) 号分组由 C 发出以后, A 将予以确认, 当A刚刚发出对第 (7) 号分组的确认而 C 还暂时没有收到这个确认的瞬间, A 和 C 在当前TCP连接上分别处于 TCP 的有限状态机的哪个状态? (请一定采用 TCP 里的状态名称) 这时A和C是否还有可能在此连接上继续收到来自对方的应用层数据?

A: CLOSE\_WAIT; C: FIN\_WAIT\_1;  
此刻, C还有可能收到来自A的数据。

考查点: 连接关闭的管理; TCP关闭的状态转移; 对半关闭的理解

3. 从主机A向主机C的80端口发起连接开始, 到它接受来自主机C的连接关闭请求并表示确认, 主机A一共向主机C发出多少应用层的数据 (即不包含IP和TCP头)? 给出你的计算依据和过程。

主机A发起连接时的序列号为0x8571C621, 建立连接后开始发送数据时的序列号为0x8571C622, 接受C的关闭连接请求时的序列号为0x8571C750, 故此间发送数据的字节数为0x8571C750 - 0x8571C622 = 0x12E=302。

考查点: SYN 占用一个序列号

备注: 如果通过分组长度和IP头、TCP头的计算获知应用层传递的数据也是可行的。但本题隐含的危险在于由于题目已经说明是“部分”分

组的摘要，这么做可能计数不完整；此外，数数的办法不可取，因为摘要并不包含分组的全部内容。

4. 注意到TCP的标志为SYN或FIN的数据段都要用掉一个序列号，哪怕不携带任何应用数据，即当主机C收到序列号为 $x$ 、标志为SYN或FIN的数据段，它将回应一个确认序列号 $\text{AcknowledgeNumber} = x + 1$ 的数据段。这是必要的吗？如果是，请说明如果SYN不占用一个序列号会带来什么问题；如果不是，请简要说明理由。

是必要的，否则无法判断是否是旧的重复的请求。（多个连接同时发生共用一个序列号几乎不可能，因为初始序列号（ISN）的选择是以系统时钟为依据的。）

考查点：对SYN 占用一个序列号的理解

备注：援引讲解TCP之间的三次握手协议，可以说明：如果连接请求不这样处理序列号，那么就必须要更复杂的方式来解决重复连接的问题；从而也能从一个反面说明TCP这样处理的必要性，使得连接握手的协议变得简单。

5. 假定中间的网络具有1 Gbps的带宽，A和C之间的往返延迟是140毫秒，现在为了最充分利用带宽资源，你可以为A和C专门设计一个新的基于滑动窗口和字节流的传输协议，那么其中通告窗口大小的字段应该有多少位（不考虑操作系统的效率）？

窗口大小  $1\text{Gbps} * 140\text{ms} / 8\text{bits/Byte} = 17.5 \text{ MB}$ ，通告字段应长  $\lceil \log_2 17.5 \times 10^6 \rceil = 25\text{bit}$ 。

考查点：窗口对流控的作用

备注：窗口实际对应一个带宽 - 时间积。

附录一：可能用到的协议号

ip	0	IP	# internet protocol, pseudo protocol number
icmp	1	ICMP	# internet control message protocol
tcp	6	TCP	# transmission control protocol
udp	17	UDP	# user datagram protocol
ipv6	41	IPV6	# ipv6

附录二：可能用到的知名端口号

echo	7/udp	# Echo
echo	7/tcp	# Echo
ftp-data	20/udp	# File Transfer [Default Data]
ftp-data	20/tcp	# File Transfer [Default Data]
ftp	21/udp	# File Transfer [Control]
ftp	21/tcp	# File Transfer [Control]
telnet	23/udp	# Telnet
telnet	23/tcp	# Telnet
domain	53/udp	# Domain Name Server
domain	53/tcp	# Domain Name Server
finger	79/udp	# Finger
finger	79/tcp	# Finger
http	80/udp	www www-http # World Wide Web HTTP
http	80/tcp	www www-http # World Wide Web HTTP