1. （Memory Consistency）(50 分) 考虑在一个实现了顺序一致性（sequential consistency）的多核处理器上，使用两个不同的线程分别（并行地）运行以下两个 C 程序。

Thread T0	Thread T1
Instr. T0.0 $X[0] = 2;$	Instr. T1.0 $X[0] = 1;$
Instr. T0.1 $flag[0] = 1;$	Instr. T1.1 $X[0] \mathrel{+}= 2;$
Instr. T0.2 $a = X[0] * 2;$	Instr. T1.2 $while(flag[0] == 1);$
Instr. T0.3 $b = Y[0] - 1;$	Instr. T1.3 $a = flag[0];$
Instr. T0.4 $c = X[0];$	Instr. T1.4 $X[0] = 2;$
	Instr. T1.5 $Y[0] = 10;$

其中 X 和 flag 已经分配在内存中，T0 和 T1 使用处理器每个核心的私有寄存器来存储变量 a, b, c 的值，对任意变量的读或写都会产生一次内存请求（memory request）。现在，内存上的所有位置以及所有变量的初始值是 1. 假设上面 C 程序的每一行代码都对应单条指令。

(a) 说明上面的 C 程序可能存在的问题。

Thread 1 will never finish.

Explanation:

The while loop in instruction T1.2 is an infinite loop, because the value of flag[0] is 1 since the beginning of the program.

(b) 在题设的处理器上运行这个程序，$X[0]$ 的最终值有哪些可能？给出所有可能值并解释出现该值时程序的运行情况。

2, 3, or 4.

Explanation:

The sequential consistency model ensures that the operations of each individual thread are executed in the order specified by its program. Across threads, the ordering is enforced by the use of flag[0]. Thread 1 will remain in instruction T1.2 until flag[0] has a value that is not 1. However, thread 1 will never finish execution. There are at least three possible sequentially-consistent orderings that lead to at most three different values of X at the end:

Ordering 1: T1.0 → T1.1 → T0.0, Final value: X[0] = 2.

Ordering 2: T0.0 → T1.0 → T1.1, Final value: X[0] = 3.

Ordering 3: T1.0 → T0.0 → T1.1, Final value: X[0] = 4.

(c) 在题设的处理器上运行这个程序， a 的最终值有哪些可能？给出所有可能值并解释出现该值时程序的运行情况。

2, 4, 6, or 8.

Explanation:

The value of a is twice the value of X[0]:

Ordering 1: T1.0 → T1.1 → T0.0 → T0.2, Final value: X[0] = 4.

Ordering 2: T0.0 → T1.0 → T1.1 → T0.2, Final value: X[0] = 6.

Ordering 3: T1.0 → T0.0 → T1.1 → T0.2, Final value: X[0] = 8.

Ordering 4: T0.0 → T0.1 → T1.0 → T0.2, Final value: X[0] = 2.

(d) 在题设的处理器上运行这个程序， b 的最终值有哪些可能？给出所有可能值并解释出现该值时程序的运行情况。

0.

Explanation:

Because the value of b depends only on the value of Y[0] (instruction T0.3). The initial value of Y[0] is 1. Instruction T1.4 will not be executed as T1 enters an infinite loop after executing instruction T1.2.

(e) 考虑在另一个没有实现内存一致性（memory consistency）的多核处理器上运行这个程序，$X[0]$ 的最终值有哪些可能？给出所有可能值并解释出现该值时程序的运行情况。

1, 2, 3, or 4.

Explanation:

Since there is no guarantee of a strict order of memory operations, as seen by different cores, instruction T1.1 could complete before or after instruction T1.0, from the perspective of the core that executes thread 0. If instruction T1.1 completes before instruction T1.0, from the perspective of the core that executes T0, instruction T0.0 could complete before or after instruction T1.0. Thus, there are at least five possible weakly-consistent orderings that lead to different values of X[0] at the end:

Ordering 1: T0.0 → T1.1 → T1.0, Final value: X[0] = 1.

Ordering 2: T1.4 → T0.0 → T1.1 → T1.0, Final value: X[0] = 1.

Ordering 3: T1.0 → T1.1 → T0.1, Final value: X[0] = 2.

Ordering 4: T0.0 → T1.0 → T1.1, Final value: X[0] = 3.

Ordering 5: T1.0 → T0.0 → T1.1, Final value: X[0] = 4.

2. （Interconnect Design）（30 分）考虑为一个包含 256 个节点的网络设计互联。每一个节点的可用资源包含一个处理器核心，一块缓存，以及一个多核处理器上分配的部分物理内存。假设这个网络的负载在空间上均衡而在时间上不均衡：即网络中的每个节点向其他节点发包的概率相等，但网络中的负载随时间变化可能变得非常高或非常低。现在，请选择最适合这个网络的互联设计，以最大化性能和能耗效率，并最小化面积和设计复杂度。

(a) Crossbar，Multistage Logarithmic 和 2D Mesh，哪个最适合作为网络拓扑？还是说都一样。请给出理由。

2D Mesh

Area: A 256x256 Crossbar would be too costly O(N2) = O(2562) = 65536. On the other hand, even though both Multistage Logarithmic and 2D Mesh are blocking, the 2D mesh is 8x less costly (O(N) = O(256)) than the Multistage Logarithmic (O(N× log2(N)) = O(256× log2(256)) = 256×8).

Latency: The crossbar has the lowest average latency between the topologyes (O(1)).

The average latency of for the 2D Mesh is twice as the Multistage Logarithmic. The average latency for the former one is O(√256) = 16, while for the later one it is O(log2(256)) = 8.

Therefore, considering both area and latency, the 2D Mesh has the best trade-off.

(b) Circuit switching，Packet switching，哪个最适合作为路由器交换方式？还是说都一样。请给出理由。

Packet switching

Since the communication traffic is uniform, and the network topology is blocking, packet switching will make better use of the switches by dynamically sharing the links.

(c) Deterministic routing，Oblivious routing，Adaptive routing，哪个最适合作为路由算法？还是说都一样。请给出理由。

Adaptative routing

Once the load changes over time from low to high, an adaptive routing can take decisions based on the number of congested or faulty channel.

(d) Bufferless，Store and Forward，Cut Through，Wormhole，Virtual Cut Through，哪个最适合作为流控策略？还是说都一样。请给出理由。

Wormhole

Since the packet size varies, all of the packet-based flow control methods (Store and Forward, Cut Through, and Virtual Cut Through) would require more buffer space than the router might require (at least the size of the biggest packet). Meanwhile, a Bufferless flow control method would increase router complexity, and produce congestions under high loads.

3 （Interconnects）（20 分）考虑网络 N 中有 2048 个处理器，现在有两种网络拓扑方案来连接这些处理器：

A. 双向环形总线（Bi-directional Ring）

B. 超立方体（Hypercube）

对于下列问题，分别计算网络 N 在**两种**拓扑下的答案：

(a) 网络 N 中的链接数量（认为每个链接是双向的）。

Ring: 2048

Hypercube: 2^N × N/2 links -> 2048 × 11/2 = 11264 links

(b) 网络 N 中的最大路由距离（即网络直径），按照跃点数（hop count）计算。

Ring: 2048 / 2 = 1024 hops

Hypercube: Log(2048) = 11 hops

(c) 若网络 N 中存在至少一个处理器不能访问其它所有处理器，则最少有多少个链接发生了故障?

Ring: 2 Links

Hypercube: Log(2048) links -> 11 links