1. （30 分）（Vector Processing）现在有一台向量计算机 VC，它的指令延迟如下：

VLD 和 VST：每个向量元素（Vector Element）开销为 50 个周期，支持 fully interleaved and pipelined。

VADD：每个向量元素开销为 4 个周期（fully pipelined）。

VMUL：每个向量元素开销为 16 个周期（fully pipelined）。

VDIV：每个向量元素开销为 32 个周期（fully pipelined）。

VRSHFA：每个向量元素开销为 1 个周期（fully pipelined）。

现在假设：

1. VC 所支持的流水线是有序流水线（In-Order Pipeline）。
2. VC 支持向量功能单元之间的链式（Chaining）操作。
3. 为支持向量元素的单周期内存访问，VC 将向量元素交错存储在内存的多个 bank 中，一个向量中的多个元素在内存中的排布如下：第一个元素映射到 Bank 0，第二个元素映射到 Bank 1，依此类推。
4. 每个 Bank 有一个 8 KB 的行缓冲区（Row Buffer）。
5. 向量元素大小为 64 位。
6. 每个 Bank 有两个端口（支持允许加载/存储操作并行），并且有两个加载/存储功能单元可用。

请根据上述条件回答以下问题：

(a) 为了使内存访问永不阻塞，Bank 数量（2 的幂次）至少是多少？（假设向量步长为 1）

64 banks (because memory latency is 50 cycles and the next power of two is 64)

(b) 假设 VC 的 Bank 数量如 (a) 所描述，且执行下列程序 P 需要花费 111 个时钟周期：

```
VLD    V1, A        // V1  ← A
VLD    V2, B        // V2  ← B
VADD   V3, V1, V2   // V3  ← V1 + V2
VMUL   V4, V3, V1   // V4_i ← V3_i × V1_i
VRSHFA V5, V4, 2    // V5_i ← V4_i >>> 2
```

请问向量长度 L（向量中的元素个数）是多少？

L = 40

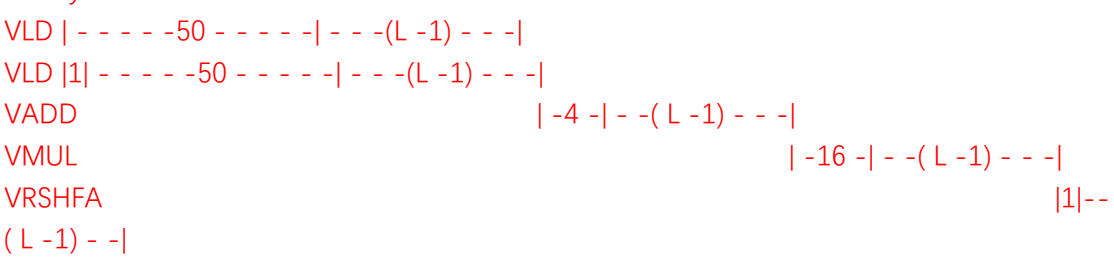
```
VLD | - - - -50 - - - - - -| - - -(L -1) - - - -|
VLD |1| - - - -50 - - - - - -|
VADD                      | -4 -|
VMUL                           | -16 -|
VRSHFA                              |1| - - - - -(L -1) - - - - -|
```

1 + 50 + 4 + 16 + 1 + (L − 1) = 71 + L = 111

L = 40

(c) 若衍生型号 VC-SE 不支持向量功能单元之间的链式操作，但仍具有 VC 的其他特性。请问 VC-SE 执行程序 P 需要多少个时钟周期？

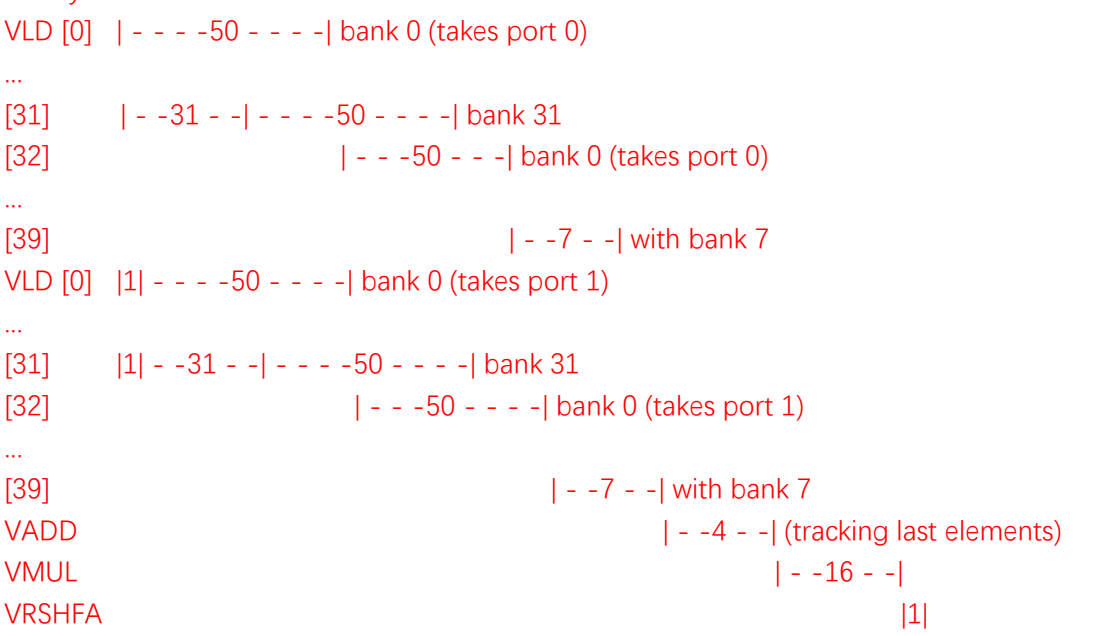
228 cycles



1 + 50 + 4 + 16 + 1 + 4 × (L−1) = 68 + 4 × L = 228 cycles

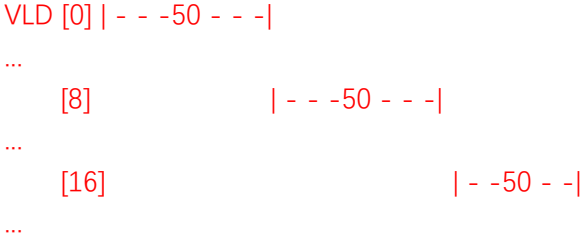
(d) 若衍生型号 VC-Mini 将内存的 bank 数量相比于 (a) 中的描述砍了一半，其余特性和 VC 保持一致。此时对内存中的向量访问会产生阻塞，所以每个 Bank 上增加了仲裁器以使得最早的访问被最先处理。请问在 VC-SE 上执行程序 P 需要多少时钟周期?

129 cycles



(B[39]: 1 + 50 + 50 + 7) + 4 + 16 + 1 = 129 cycles

(e) 若 VC-Tiny 进一步缩减 bank 数量（但始终是 2 的幂次），使得执行完程序 P 需要 279 个时钟周期。请问 VC-Tiny 上有多少个 Bank 数量?



[24] | - -50 - -|

...

[32] | - -50 - -|

...

[39] | - -7 - -|

VLD [39] |1|

VADD | - -4 - -|

VMUL | - -16 - -|

VRSHFA

|1|

5 × 50 + 7 + 1 + 4 + 16 + 1 = 279 cycles => 8 banks

(f) 若 VC-Ultra-100 支持多核处理，其具有 4 个向量处理器，共享一个内存系统，bank 数量是 VC 的 4 倍。现在在 VC-Ultra-100 的每个核心上运行测试程序，发现每个核心消耗的时间甚至比单核 VC 配 1/4 数量的 bank 还要多。请问为什么会出现这种情况？

Row-buffer conflicts (all cores interleave their vectors across all banks).

(g) 若 VC-Ultra-200 只改变 VC-Ultra-100 的共享内存架构，请问需要怎么做缓解 (f) 中出现的情况？

Partition the memory mappings, or using better memory scheduling.

2. （25 分）（SIMD Processing）现在希望设计一个能够支持向量长度为 16 的 SIMD 处理器，考虑以下两个方案：传统的向量处理器和传统的阵列处理器。

(a) 哪种方案芯片面积最大？请给出理由。

An array processor requires 16 functional units for an operation whereas a vector processor requires only 1.

(b) 假设两种处理器的加法操作延迟都是 5 周期，且加法器是 fully pipelined 的，请计算并给出理由：

i. 考虑向量长度是 1，两种处理器执行 VADD 操作花费的时钟周期？

The traditional vector processor:
5 cycles

The traditional array processor:
5 cycles

ii. 考虑向量长度是 4，两种处理器执行 VADD 操作花费的时钟周期？

The traditional vector processor:
8 cycles (5 for the first element to complete, 3 for the remaining 3)

The traditional array processor:
5 cycles

iii. 考虑向量长度是 16，两种处理器执行 VADD 操作花费的时钟周期？

The traditional vector processor:
20 cycles (5 for the first element to complete, 15 for the remaining 15)

The traditional array processor:
5 cycles

3. （25 分）（GPUs and SIMD）我们将程序在 GPU 上运行的 SIMD 利用率定义为：某程序运行期间繁忙的 SIMD 通道（busy SIMD lanes）数量比上该程序运行使用的线程数量。现在在 GPU 上运行程序 P，每个线程执行程序 P 中循环的单次迭代（包含两条指令）：

```
for (i = 0; i < N; i++) {
    if (A[i] % 3 == 0) {      // Instruction 1
       A[i] = A[i] * B[i];    // Instruction 2
    }
}
```

假设数组A和B的数据已经位于向量寄存器中(此时执行程序P中不需要加载和存储操作)。假设 GPU 的一个 wrap 有 32 个线程，GPU 有 32 个 SIMD 通道。假设每条指令花费的时间相同。

(a) 用 N 表示执行程序 P 所需要的 wrap 数量。

Ceiling(N/32)

(b) 假设整数数组 A 具有重复模式，24 个 1 后跟 8 个 0 重复出现，而整数数组 B 具有另一种重复模式，48 个 0 后跟 64 个 1。此时程序 P 的 SIMD 利用率是多少？

0.625

((24+8×2)/(32×2)) = 40/64 = 0.625

(c) 程序 P 的 SIMD 的利用率有可能达到 100%么？如果有可能，请给出数组 A 和 B 满足的条件；如果不可能，请给出理由。

Yes, Starting from A[0], consecutive 32 elements of A should be either divisible by 3 or not divisible by 3.

B can be any array of integers.

(d) 程序 P 的 SIMD 的利用率有可能达到 56.25%么？如果有可能，请给出数组 A 和 B 满足的条件；如果不可能，请给出理由。

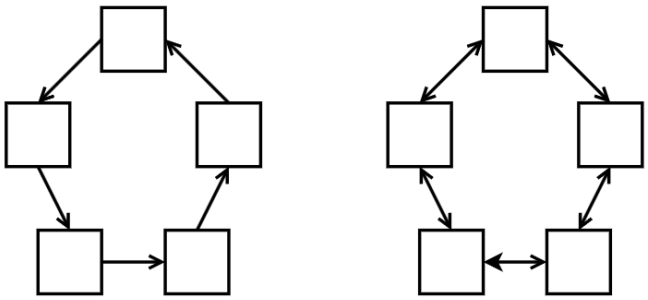
Yes, 4 out of every 32 elements of A are divisible by 3

B can be any array of integers.

(e) 程序 P 的 SIMD 的利用率有可能达到 50%么？如果有可能，请给出数组 A 和 B 满足的条件；如果不可能，请给出理由。

The minimum is when only 1 out of every 32 elements of A is divisible by 3. This yields a 51.5625% usage.

4. （10 分）（Interconnects）下列两张图展示了包含 n 个节点的两种环形拓扑。



a) Uni-directional Ring　　b) Bi-directional Ring

假设网络满足下列条件：

1. n 是一个奇数。
2. 数据包可以在 1 个周期内从一个节点移动到相邻节点。
3. 路由机制使用从源节点到目标节点的最短路径。
4. 通信模式是均匀的（即每个节点向每个其他节点发送数据包的概率相等）。
5. 没有竞争（即数据包在每个周期总是可以通过最短路径向其目标节点移动）。

请回答下列问题：

(a) 大小为 n 的单向环的平均延迟是多少？请展示你的计算过程。

Avg(1 + 2 + 3 + · · · + n−1) = (n-1+1)(n−1)/2/(n-1) = n/2

(b) 大小为 n 的双向环的平均延迟是多少？请展示你的计算过程。

2 * Avg(1 + 2 + 3 + · · · + (n−1)/2) = 2(1+(n-1)/2)/2/2= (n+1)/4

5. （10 分）(Interconnects) 考虑连接一个包含2^N处理器的系统，使用下面三种拓扑结构：

i. $\sqrt{2^N} \times \sqrt{2^N}$ 2D mesh

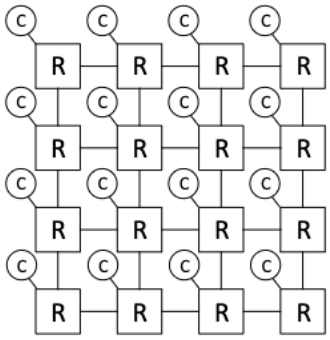
ii. $\sqrt{2^N} \times \sqrt{2^N}$ 2D torus

iii. Hypercube

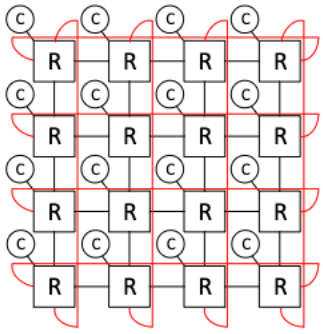
请回答下列问题：

(a) 绘制 N=4 时候每种拓扑对应的网络结构图（适当使用省略号简化绘图）

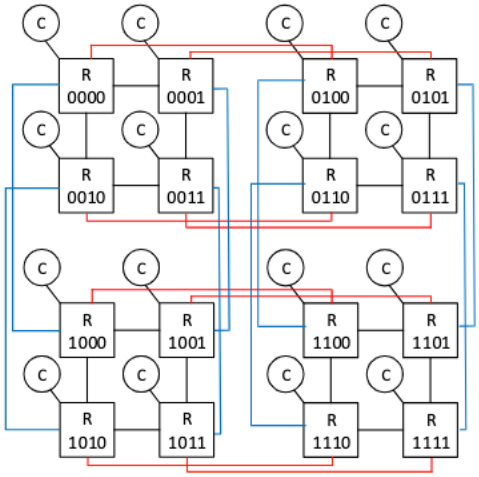
2D Mesh:



2D Torus:



Hypercube:



(b) 计算 N=8 时候每种拓扑的网络链路数量（每个链接时双向的）。

2D mesh: $2 \times (\sqrt{2^N} - 1)(\sqrt{2^N})$ links $\rightarrow 2 \times 15 \times 16 = 480$ links

2D torus: $2 \times (\sqrt{2^N})(\sqrt{2^N})$ links $\rightarrow 2 \times 16 \times 16 = 512$ links

Hypercube: $2^N \times N/2$ links $\rightarrow 256 \times 8/2 = 1024$ links

(c) 计算 N=8 时候每种拓扑中的输入/输出端口数量（包括 router 的端口）。对于 irregular network，还需要给出每种类型 router 的端口数量。

2D mesh: (4+1) inputs/outputs, (3+1) inputs/outputs, and (2+1) inputs/outputs

2D torus: (4+1) inputs/outputs

Hypercube: N+1 inputs/outputs => 9 inputs/outputs