一个程序员，写了如下C语言代码。他想在多核处理器上运行该程序，用两个线程，每个线程占一个核。处理器采用顺序一致性协议，变量X和flag存储在内存中，变量a和b存储在寄存器中。所有内存初始化为0。假设每一行C代码代表一条指令。

|  |  |
| --- | --- |
| Thread T0 | Thread T1 |
| 指令T0.0 X[0] = 1; | 指令T1.0 X[0] = 0; |
| 指令T0.1 X[0] += 1; | 指令T1.1 flag[0] = 1; |
| 指令T0.2 while(flag [0]] == 0); | 指令T1.2 b = X[0]; |
| 指令T0.3 a = X[0]; |  |
| 指令T0.4 X[0] = a\*2; |  |

1. 变量a的最终值可能是多少？ 解释原因

2. 变量X[0]的最终值可能是多少？解释原因。

3. 变量b的最终值可能是多少？解释原因。

4. 假设该程序员想让变量a和b的取值在程序执行完毕时是相同的，那么在保留T1.1和T0.2两条指令的前提下，需要对原始程序如何做最小的改动（填写下表）？（提示：可以利用更多的flags）

|  |  |
| --- | --- |
| **Thread T0** | **Thread T1** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

一个程序员，写了如下C语言代码。他想在多核处理器上运行该程序，用两个线程，每个线程占一个核。处理器采用顺序一致性协议，变量X和flag存储在内存中，变量a和b存储在寄存器中。所有内存初始化为0。假设每一行C代码代表一条指令。  
Thread T0   
指令T0.0 X[0] = 1;

指令T0.1 X[0] += 1;

指令T0.2 while(flag [0]] == 0);

指令T0.3 a = X[0];  
指令T0.4 X[0] = a\*2;

Thread T1

指令T1.0 X[0] = 0;

指令T1.1 flag[0] = 1;

指令T1.2 b = X[0];

1、变量a的最终值可能是多少？ 解释原因

2、变量X[0]的最终值可能是多少？解释原因。

3、变量b的最终值可能是多少？解释原因。

4、假设该程序员想让变量a和b的取值在程序执行完毕时是相同的，那么在保留T1.1和T0.2两条指令的前提下，需要对原始程序如何做最小的改动（填写下表）？（提示：可以利用更多的flags）

Thread T0 Thread T1