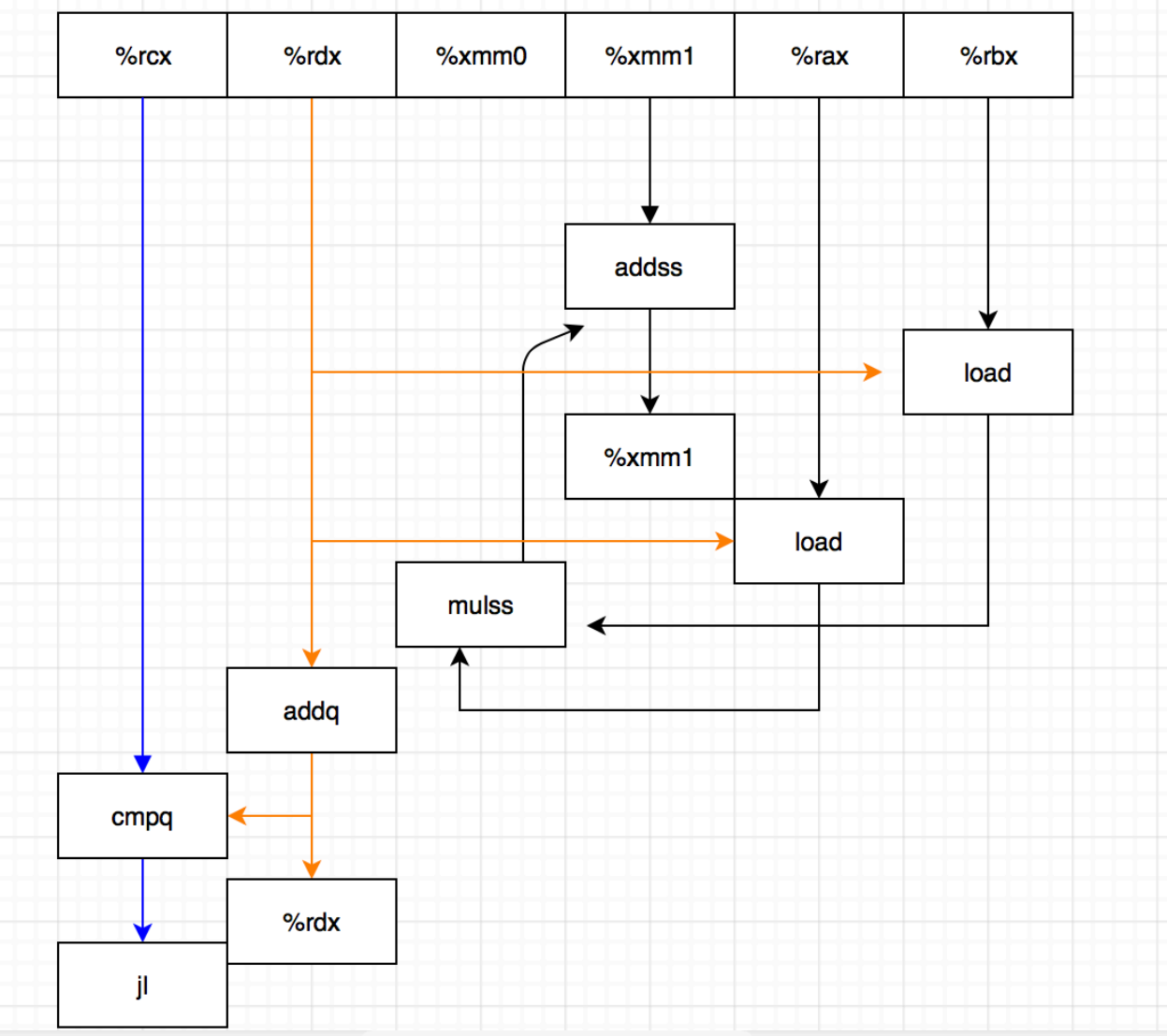
Solution:
1.
A.



B. The critical path is the one with the most activities which should be finished on time.
So the lower bound of CPE is according to the critical path:
The critical path is the one with one adds and one mulss, float point latency of add is 3, multiplication is 4.
The amount of instructions is 3.
So answer is (4+3)/3=2.33

C. Same critical path, one multiplication one add of integer:
So answer is (1+3)/3=1.33

D. The loads have no dependency on previous iteration, so they can pipeline in advance.

For example, i is from 0 to 3 then sum = ((((sum+udata[0]*vdata[0])+ (sum+udata[1]*vdata[1]))+ (sum+udata[2]*vdata[2]))+ (sum+udata[3]*vdata[3])).
When it is at most executed 4 cycles, i is executed 5 times.
So CPE is the latency of add of FP.

2.
A. Pipeline help this edition quick than 5.15's edition. However, every clock cycle the load can only initiate one load operation, but the mul operator which also needs two cycles to perform needs to get two values in order to go on.

3.
A. Although the 4 accumulators have no dependency each other and it helps loads to pipeline, it depends on the maximum throughput of a relevant functional unit. In order to get one element we need 2 loads, and within a unit it can only load one value per clock cycle, but we need 2 element to do the FP multiplication which perform every two clock cycles.

4.
When VECTOR_SIZE increases, separately the three spent more time. However, 5-15 always spent more time than 5-16 and 5-17. And when the number becomes very big, it seems that 5-16 becomes the one spent the minimum time. The time spent from much to less: 5-15, 5-17, 5-16.