1) (POINTS 25/30)

Consider the following snippet of code running on 4-ways out-of-order superscalar processor.
Initially, R1=0x1000, R3=0x3000, R7=0x0003 and the other registers contain zero.

```
lab1:   LW   R2,0(R1)
        MUL  R4,R2,R2
        SW   R4,0(R1)
        ADDI R1,R1,4
        BNE  R2,R0,lab1
```

Working hypothesis:
* the fetch, decode and commit stages are 4 instructions wide
* the instruction window has 8 slots
* we have 8 physical registers in the free pool
* the reorder buffer has 10 entries
* the integer multiplier has 4 stages
* the load/store queues have 3 slots each and a common effective-address calculation unit
* there are 4 ALUs for arithmetic and logic operations and for branching
* an ALU/BRANCH performs its operation in the same cycle when the operation is issued
* reads require 1 clock cycle (after the addressing phase)
* the register file has 4 input- and 4 output-ports
* there are 9 logical registers (including R0 which is hardwired to 0)
* the store operation leaves the issue stage as it is inserted in the store queue
* when the X stage finds a free slot in the LQ/SQ, also a slot in the I stage is reserved
* branches are predicted taken

In order to calculate the total cycles needed to execute three iterations of the above loop on such machine,
complete the following chart until the end of the THIRD iteration of the code fragment above, including the
renamed stream the precise evolution of the free pool of the physical registers (the register map), the
Instruction Window (IW), the Reorder Buffer (ROB) and the Load Queue (LQ) and Store Queue (SQ).
Note: Ci, Cj, Ck are the cycles when the corresponding physical registers are available in IW.

Iter.	Instruction	F Fetch (clock)	D Decode (clock)	P disPatch (clock)	I Issue (clock)	X eXecute (clock)	W Write-back (clock)	C Commit (clock)	Renamed Stream	Instruction Window Pi Pj Pk I	Reorder Buffer Ci Cj Ck Ri Pi,old Cplt	Load Queue Store Queue
1	LW R2,0(R1)	0							P2,0(P1)	P2 P1 - 0	0 0 0 R2 - -	
...	...											
...	...											

2) (POINTS 5/30) On a Linux system, write the SINGLE command line to perform at the BASH shell prompt the
following operation (please note that no intermediate files should be used:
   • The file 'data1.txt' contains a list of alpha-numerical values to be used as input
   • The file 'data2.txt' should contain a list of the lines which start with "al"
   • The extracted list should be directed to a file data3.txt