1. **Who does it?** (9 pts)
For each action listed below, indicate which of the listed entities takes the action. The lists are the same for all questions. There could be more than one answer for each action.

(a) After an I/O interrupt, it saves the CPU state (PC, registers, condition codes, etc.) in the PCB (process control block).

- clock interrupt signal
- clock interrupt handler
- I/O interrupt signal
- I/O interrupt handler
- OS scheduler
- user process

**Why?** This is the first thing an interrupt handler takes care of. The I/O interrupt handler handles an I/O interrupts.

(b) changes a process‘ state in the PCB from ‘running‘ to blocked

- clock interrupt signal
- clock interrupt handler
- I/O interrupt signal
- I/O interrupt handler
- OS scheduler
- user process

**Why?** When a process goes from running to blocked, it means it has voluntarily requested service from the OS. The currently running process must wait/block for the service to be completed and thus the OS scheduler (or OS service routines) will block it. (Note that no interrupt handler is responsible for this task since they are only activated when someone else’s I/O interrupt the current process.

(c) does not use the fetch/execute cycle to accomplish its task(s)

- clock interrupt signal
- clock interrupt handler
Why? the first two are hardware signals and do not involve executable code. The others all involve code executed through the normal fetch/execute cycle.

2. Frequency of events (10 pts)
For the following pairs of events, write 'less than,' 'greater than,' 'approximately equal,' or 'cannot predict' to describe the relative frequency of the two events. You can use the abbreviations <, >, =, and ? respectively. Explain each answer in one sentence.

(a) an interrupt occurred  ____< or <=____ OS scheduler is called
OS scheduler is always called after an interrupt to choose a new process to run, but it is also called other times (e.g. when processes request services such as I/O or memory)

(b) an interrupt occurred  ____=____ an interrupt handler is executed
Every interrupt causes control to pass to the corresponding interrupt handler.

(c) process A is in blocked state  ____<____ process A is in ready state
Every blocked process goes to ready state but there are other ways to get into ready state, e.g. from new or from running. Look at the 5 state diagram.

(d) process A is in new state  ____=____ a process A is in terminated state
A process is in new state once and in terminated state once.

(e) a timer interrupt occurs  ____?__ or >_ an I/O interrupt occurs
This is entirely dependent on the type of workload running on the system. CPU-bound jobs are interrupted primarily by timer interrupts. I/O-bound jobs have a lot of I/O interrupts. But the relationship depends on the mix. If you argued that there are likely to be more timer interrupts than I/O interrupts, that is OK.