Assignment 4
PC ISRs

due
Monday, November 9 1998
12:30 PM

Question 1

Write an 8088 assembly-language program that does the following:

- prints your name and student number. You may use DOS interrupt 21H function 09H (AH should be set to 09H and DS:DX should point to string that is terminated with a '$' character).
- initializes a count variable to zero.
- saves the previous value of the keyboard interrupt vector by PUSHing the old CS and IP values on the stack (not by using MOV instructions as in the example in the lecture notes).
- sets up the keyboard interrupt vector to point to an ISR that you'll also write (see below).
- loops continuously until the count variable becomes equal to 2. The value of the count should not be changed by instructions in the loop.
- restores the previous interrupt vector and returns control to DOS using interrupt 20H.

Write an interrupt service routine for the keyboard interrupt that does the following:

- saves and initializes any registers that need to be saved on the stack of the interrupted program (not in a fixed memory location).
- reads a byte from I/O port address 60H. This is the keyboard data port.
- if the value read is 82H (the scan code generated by releasing the '1' key) increments the count variable.
- gives an EOI command to the PIC, restores all registers that were saved on entry to the ISR and returns from the interrupt.

Note that your program *must* use an ISR that reads the keyboard port and tests the scan code directly. *Do not* use BIOS or DOS routines to read the keyboard.

Assemble the main program and the ISR into an executable .COM file and test it by running it and then pressing various key sequences. Your program should terminate only after the '1' key has been pressed twice.

Note that debugging ISRs can be difficult because often: (1) the exact flow of control is unclear, (2) errors will cause the computer to crash and require a reboot, and (3) debuggers can't trace into the ISR because interrupts are disabled. You may find it more efficient to start with a simple program and gradually add functionality. You can use the INT instruction to invoke your ISR for testing.

As usual, copies of others' solutions to this assignment will receive a mark of 0.