

Homework 3: GCC conventions

Answer all questions. Submit your hand-written answer sheets to the instructor before the lecture begins.

1. What is the advantage of using callee and caller-saved registers? Why can't all the registers be either callee-saved or caller-saved?
2. Why do we need to save all the registers on the stack on an interrupt? Can we only save callee-saved registers?
3. Do you think that saving register on the stack also works with recursive interrupts (a recursive interrupt mean an interrupt handler can also be interrupted)? Justify your answer.
4. Look at the implementation of “`printf`” in Pintos source code. `printf` takes a variable number of arguments. A function that receives a variable number of arguments must declare the name of at least one parameter. Given the name of a parameter, “`va_start`” routine can retrieve additional parameters. Do you think the compiler can retrieve the first argument, if the arguments are passed in the reversed order, i.e., the argument just after the return address is the nth argument? Justify your answer.