FYI
If there are any register usage conventions, I will tell you what they are
If you need to know the size of a type in any language, I will tell you what it is, if necessary

SUBJECT MATERIAL

IA32 (all instructions covered except set statements)
Y86 (all instructions)
Differences between the two assembly languages
Need to be able to recognize/evaluate *correct* code for both IA32 and Y86
i.e. what is in regs, memory, stack info, result of function, etc
Know when are condition code bits set and how used (jump/cmovs)
C code to equivalent assembly (both IA32 and Y86) and vice versa
Addressing modes
CISC/RISC
Big/little endian
Load/store architecture
Define disassembler – what does it do; when use and why
Encoding IA32 instructions (opcode given)
Stack Stack Stack
Define/differentiate between relocatable and executable object files
What do linkers do?
Symbol table info:

- Symbol definitions are stored (by compiler) in a “symbol table”
- A symbol table is an array of structs
- Each entry includes name, size, and location of symbol
- Linker associates each symbol reference with exactly one symbol definition

GIVEN
Encoding scheme pages for both IA32 and Y86; plus IA32 register page