Hints and tips:

* Most of the questions in the final are closer to midterm 2 than midterm 1. This is because I don’t need to ask you specifically how data look like in memory or how to convert from binary to hex, these topics are built-in to the midterm 2 type of questions.
* The T/F questions are on Modules 1, 2, 3. They are to catch the theories that can’t be easily shown by coding such as cache, instruction cycles, memory management, etc.
* To prepare for the final, review the class exercises and lab assignments of modules 4 – 8 and the midterms. If you can do them, you should be okay with the final.

1. T/F questions. Similar to the quizzes and midterm 1.

Example: the value 0A0h is a negative value if it’s a signed WORD.

F

2. Short answers

Example 1: Give an example of a CISC instruction from the Intel instruction set. Explain why it's a CISC instruction.

scasb is a cisc instruction and it is cisc and not risc since it does multiple things in one line of code

Example 2: Step through the instructions and show change in the general purpose registers.

mov eax, -1 ; eax = FFFF FFFF

shr eax, 2 ; eax = 3FFF FFFF (SHR = arithmetic)

cdq ; eax = FFFF FFFF edx = 0000 0000

mov ebx, 1000h ; eax = FFFF FFFF edx = 0000 0000 ebx = 0000 1000

idiv ebx ; eax = 0003 FFFF edx = 0000 0FFF ebx = 0000 1000

Example 3: The user wants to do the task that's described in the comment.  
If there is error in the instructions below so that the task can't be done, correct the error.

mov eax, offset arr ; arr contains signed words

mov edx,[eax] ; want: edx = first value of arr

; error: needs to be: move edx, WORD PTR [eax]

mov [eax + 2], 5 ; want: arr[2] = 5

; error: needs to be: mov WORD PTR [eax+4],5

3. Writing code.

Example 1: Write code to implement the following loop. All variables are unsigned bytes. Catch all possible errors with jumps to error label E1.

while (var1 > var2)

var2 = var2 \* 5

Example 2: The following question is longer than the typical exam question. It could be 3 different exam questions, to give you more practice, and in the real exam you would only need to do one part of this question.

Write a procedure to copy str1 to str2. Both strings are text strings that are null terminated. The procedure returns 1 if all of str1 can be copied to str2, and returns 0 if str1 cannot be copied to str2 because str2 is smaller in size. Use string instructions as needed.

The procedure call is

sub esp, 4 ; for return value of 1 or 0

push OFFSET str1

push OFFSET str2

call stringCopy