Question 1 [1 points]: Suppose the 32-bit value \(0x12345678\) is stored at address \(0x24\) of a little-endian computer. What byte is stored at address \(0x25\)?

A) A known value not listed here  
B) \(0x43\)  
C) An unknown value because the number does not overlap address \(0x25\)  
D) \(0x56\)  
E) \(0x34\)  
F) \(0x65\)

Answer: D

Question 2 [1 points]: \(a\) is the 8-bit value \(00110010\) and \(b\) is the 8-bit value \(10101110\). What is \(a \land b\)?

A) \(10111110\)  
B) \(11011100\)  
C) \(00100010\)  
D) \(10011100\)  
E) \(11100000\)  
F) \(00110010\)  
G) \(10101110\)  
H) none of the above

Answer: D

Question 3 [1 points]: The code \((a == b) || (a == c) || (a == d)\) implements

A) MUX  
B) Equality  
C) Decoder  
D) Set membership  
E) Adder  
F) None of the above

Answer: D
Question 4 [1 points]: Suppose the 32-bit value 0x12345678 is stored at address 0x24 of a big-endian computer. What byte is stored at address 0x23?

A 0x65  
B 0x34  
C 0x43  
D A known value not listed here  
E An unknown value because the number does not overlap address 0x25  
F 0x56  

Answer: E

Question 5 [1 points]: Write 0x19 in binary. Do not include any leading 0s.

Answer: 11001

Question 6 [1 points]: Assume %eax contains the number 30 and %ecx contains the number 50 (both presented in decimal). What address is read by the x86 (or y86) operation movl 4(%eax), %ecx?

A 54  
B 50  
C 34  
D 120  
E 200  
F 26  
G 46  
H 30  

Answer: C

Question 7 [1 points]: Suppose that the value stored in byte b of memory is b + 16. What is the value in %eax after running the Y86 command mrmovl 0x24, %eax?

A 0x24  
B 0x40  
C 0x34  
D None of the above  

Answer: D

Question 8 [1 points]: What is a “caller-save” register?

A A special register that saves who called the procedure  
B A program register that the called procedure may not modify  
C A program register that the called procedure may modify  
D None of the above  

Answer: C
Question 9 [1 points]: Suppose we are executing `addl %eax, %ebx`. Which two named values in the sequential Y86 architecture are fed into the ALU as its operands?

A  %eax and %ebx
B  valA and valB
C  0 (%eax is register number 0) and 3 (%ebx is register number 3)
D  regA and regB
E  None of the above

Answer: B

Question 10 [1 points]: The named value valM in the sequential Y86 architecture is the value read from memory. It is the input for a register in `mrmovl` and which operation in the list below?

A  subl
B  nop
C  irmovl
D  jne
E  ret input to PC
F  halt
G  call
H  rrmovl
I  popl input to argument register
J  rmmovl
K  pushl

Answer: E or I

Question 11 [1 points]: Why does Y86’s `cmovl` (the conditional move) use the ALU?

A  To compute if the value ought to move
B  Trick question – `cmovl` doesn’t use the ALU
C  To compute where to move the value from
D  To compute where to move the value to
E  To compute the value being moved

Answer: B

Question 12 [1 points]: What is two’s-compliment 11…1100 in decimal?

Answer: −4

Question 13 [1 points]: We write labels in our assembly, but they are not part of the ISA (either x86 or Y86). Why not?

A  They’re turned into addresses when we assemble the files
B  They’re like comments: they only have meaning to us, not to the computer
C  They just change how other instructions are encoded
D  The assembler turns them into other instructions as a pre-processing stage

Answer: A
Question 14 [1 points]:  a is the 8-bit value 00110010 and b is the 8-bit value 10101110. What is a - b?
A 10101110  
B 10011100  
C 11100000  
D 10111110  
E 11011100  
F 00100010  
G 00110010  
H none of the above  
Answer: H

Question 15 [1 points]:  We can encode the C command a ? b : c, where all three variables are of type int, as
A (a & b) | ((!a) & c)  
B ((~!a) & b) | ((~!a) & c)  
C ((!a) & b) | ((!a) & c)  
D Any of the above  
E None of the above  
Answer: B

Question 16 [1 points]:  The code (a && b) || (!a && !b) implements
A MUX  
B Adder  
C Set membership  
D Equality  
E Decoder  
F None of the above  
Answer: D

Question 17 [1 points]:  What is binary 110010 in decimal?
Answer: 50

Question 18 [1 points]:  Which of the following x86 operations reads values from two registers?
A addl %eax, %ebx  
B movl %eax, (%ebx)  
C movl %eax, %ebx  
D movl %eax, (%ebx,%ecx)  
Answer: B or A

Question 19 [1 points]:  Assume x is the most positive signed integer. What is x + 1?
A $\infty$  
B 0  
C $-1$  
D $-x$  
E none of the above  
Answer: E
Question 20 [1 points]: a is the 8-bit value 00110010 and b is the 8-bit value 10101110. What is a + b?
A 10101110
B 11011100
C 10011100
D 10111110
E 11100000
F 00100010
G 00110010
H none of the above

Answer: E

Question 21 [1 points]: In general, which of the following is slowest?
A moving from one register to another
B comparing two numbers to decide where to jump
C doing division
D doing addition
E accessing memory

Answer: E

Question 22 [1 points]: Which of the following features of Y86 makes it like a RISC architecture?
A Math ops only function on registers
B Parameters are passed on the stack
C Instructions have variable lengths
D All of the above are RISC-like
E None of the above are RISC-like

Answer: A

Question 23 [1 points]: The bias of any IEEE-style floating point number is $2^{e-1} - 1$, where $e$ is the number of exponent bits. If largest normalized binary number a particular IEEE format can hold is 1111.11, how many exponent bits does this format have?
A 3
B 5
C 1
D 2
E 4
F 9
G 8

Answer: A

Question 24 [1 points]: What is binary for the most negative five-bit two’s-compliment number? Answer in binary (i.e., with 5 bits, each either a 1 or a 0)

Answer: 10000
Question 25 [1 points]: Suppose the 32-bit value 0x12345678 is stored at address 0x24 of a big-endian computer. What byte is stored at address 0x25?

A  0x56
B  A known value not listed here
C  0x65
D  0x43
E  An unknown value because the number does not overlap address 0x25
F  0x34

Answer: F

Pledge:
On my honor as a student, I have neither given nor received aid on this exam.

Your signature here