**CS 3173 Basic Computer Architecture Final Exam 100 points Name:**

**One exam submission per student. Microsoft Word format only.**

**Due December 9. No late exams. Email your completed exam to me at** [**harringp@nsuok.edu**](mailto:harringp@nsuok.edu)

**9 Questions**

**Part 1: Programming in Java:**

1. *(30 points)*

Write a Java program to get input from the user for the present state, input, and then show the next state based upon the excitation table for the bit counter shown below:

**Present State Input Next State**

**A x A**

--------------------------------------------------------------------

0 0 Undefined

0 1 1

1 0 0

1 1 Undefined

1. **import** java.util.Scanner;
3. **public** **class** Final1 {
4. **public** **static** **void** main(String[] args) {
5. Scanner scan = **new** Scanner(System.in);
6. **boolean** a = **false**;
7. **boolean** input = **false**;
8. **boolean** nexta = **false**;
10. System.out.println("\nPlease enter \"true\" or \"false\"");
11. **for** (**int** i=0;i<2;i++) {
12. **if** (i==0) System.out.print("Present state of A = ");
13. **else** **if** (i==1) System.out.print("Input = ");
14. String in = scan.nextLine().toLowerCase();
15. **if** (in.equals("false")) {
16. **if** (i==0) a = **false**;
17. **else** **if** (i==1) input = **false**;
18. }
19. **else** **if** (in.equals("true")) {
20. **if** (i==0) a = **true**;
21. **else** **if** (i==1) input = **true**;
22. }
23. **else** {
24. System.out.println("\nPlease enter \"true\" or \"false\"");
25. i--;
26. }
27. }
29. **if** ((a==**false**)&&(input==**true**)) nexta = **true**;
30. **if** ((a==**true**)&&(input==**false**)) nexta = **false**;
32. **if** ((a==**false**)&&(input==**false**)) System.out.println("Next A: undefined");
33. **else** **if** ((a==**true**)&&(input==**true**)) System.out.println("Next A: undefined");
34. **else** System.out.println("Next A: "+nexta);
35. }
36. }
37. *(30 points)*

Write a Java program to use the Boolean data type for A and B and get input from the user. Use statements representing the two logic gates below:

(A || B) || (A && B) (A || B) && (A || B)

1. **import** java.util.Scanner;
3. **public** **class** Final2 {
4. **public** **static** **void** main(String[] args) {
5. Scanner scan = **new** Scanner(System.in);
6. **boolean** a = **false**;
7. **boolean** b = **false**;
9. System.out.println("\nPlease enter \"true\" or \"false\"");
10. **for** (**int** i=0;i<2;i++) {
11. **if** (i==0) System.out.print("A = ");
12. **else** **if** (i==1) System.out.print("B = ");
13. String in = scan.nextLine().toLowerCase();
14. **if** (in.equals("false")) {
15. **if** (i==0) a = **false**;
16. **else** **if** (i==1) b = **false**;
17. }
18. **else** **if** (in.equals("true")) {
19. **if** (i==0) a = **true**;
20. **else** **if** (i==1) b = **true**;
21. }
22. **else** {
23. System.out.println("\nPlease enter \"true\" or \"false\"");
24. i--;
25. }
26. }
28. System.out.println("(A||B)||(A&&B) = "+((a||b)||(a&&b)));
29. System.out.println("(A||B)&&(A||B) = "+((a||b)&&(a||b)));
30. }
31. }

**Part 2: Questions and Problem Solving:**

1. *(5 points)* What is the largest address space for a CPU with a 128-bit address bus?

**2^128 bytes**

1. *(5 points)* List and describe the four general purpose registers.

**AX – Accumulator Register – does math, logic, and data transfer**

**BX – Base Register – uses as a base address for pointing at things such as data arrays**

**CX – Counter Register – counts the index of loops**

**DX – Data Register – used with AX register for special functions**

1. *(5 points)* What is the difference between a latch and a flip-flop?

**A latch is active-low; a flip-flop is active-high.**

1. *(5 points)* Use a bitwise lower nibble mask to isolate the lower four bits (lower nibble) of the following number: 10100111

**10100111 AND 00001111 = 00000111**

1. *(5 points)* Use bitwise OR with a bitwise mask to set bits 3 and 4 to one in the following number: 10100111

**10100111 OR 00011000 = 10111111**

1. *(5 points)* Use bitwise XOR with a bitwise mask to toggle bits 0 and 2 in the following number: 10100111

**10100111 XOR 00000101 = 10100010**

1. *(10 points)* Use the 2’s complement checksum to calculate the resulting value to check for errors in the following binary number: 10100111

**10100111 + 01011001 = (1)00000000**