H212 Early Evaluation Exam Spring 2019 Name/Username:_____

Simplify[1] the following expressions where `b` is a boolean and `n` is an integer:

1. `b == false`
2. `b != false`
3. `b && true`
4. `b || true`

5. `b && !b`
6. `b || !b`
7. `n < 3 || n > 5`
8. `n < 3 && n > 5`

9. `b = n++ > n;`
10. `if (n < 10) { b = true; } else { b = n > 20; }`

11. `b = (n < 10) ? (n > 5) : (n < 20)`

12. `b = (n < 10) ? (n > 5) : true ? false : (n < 20)`

13. `b = (n < 10) ? (n > 5) : false ? true : (n < 20)`

Evaluate the following expressions (where this applies assume `n` is an integer):

14. `7%9`
15. `9%7`
16. `7/9`
17. `9/7`
18. `1/2*3`
19. `3*1/2`

20. `5-2+4`
21. `5-(2+4)`
22. `3*(1/2)`
23. `n==n++`
24. `n==++n`
25. n++ < n

Are these two code fragments[2] equivalent or not (explain):

26. (a) `if (n == 8) n = n + 3; else n = 5;`
    (b) `if (n == 8) n = n + 3; if (n != 8) n = 5;`

27. If m and n are integers[3] can you simplify this expression (and if so what does it simplify to)?

```
m / n * n + m % n
```

Evaluate the following expressions:

28. `"1" + 2 + 3`
29. `1 + "2" + 3`
30. `1 + 2 + "3"`
31. `'5' + 3`

32. `'5' - 3`
33. `"whatever".substring(2, 5)`
34. `"whatever".substring(4)`

35. `"whatever".substring("what".length(),"whatever".length() - 1)`

---

[1] If you can't simplify an expression please say so and provide a clearer, equivalent expression.

[2] Assume n is an integer.

[3] In this case assume n is not zero.

36. Make up an example in Java that demonstrates what is known as "the dangling[4] else problem" using the following statement: "A student with a GPA of at least 1.5, but less than 2, is on probation. With less than 1.5, the student is failing."

Before After

37. Rewrite the following `do` loop as a `while` loop:

```
int count = 0; do { System.out.println( ++count ); } while (Math.random() < 0.5);
```

38. If x is 6 before the next fragment gets executed what value does y have at the end?

```
if (x > 3) { if (x <= 5) y = 1; else if (x != 6) y = 2; } else y = 3;
```

What are the types of each of the following Java expressions:

39. `Math.sqrt(2)` 40. `System.out` 41. `3` 42. `'3'` 43. `"3"`

44. `3l` 45. `3.14` 46. `3.14f` 47. `1 + '3'` 48. `1 + "3"`

49. Does this program fragment ever terminate? If so, how many iterations does it take and what is the value of x at the end? If it doesn't, why doesn't it?

```
int i = 10;
while (i > 0) ; {
  i = i – 1;
}
```

For each of the following code fragments determine the value of y at the end:

50. `int x = 18, y = 10; if (x < 10) { if (x > 5) y = 1; } else y = 2;`

51. `int x = 18, y = 10; if (x < 10) if (x > 5) y = 1; else y = 2;`

52. `int y; boolean x = false; if (true) y = 2; else y = 1;`

53. `boolean y; if (false) y = true; else y = false;`

Evaluate:

54. `false && false || true` 55. `false && (false || true)`

[4] When something dangles there is a before and an after to the process. Make sure you pay attention to that.