Concept: comparisons

1. Which of the following is not a valid Boolean expression?
   (a) x != y
   (b) x = y
   (c) x > y
   (d) x <= y

2. Which of the following is a valid Boolean expression?
   (a) x + y
   (b) x = y
   (c) x != y
   (d) x += y

3. Which operator returns false when two things are not equal.
   (a) !=
   (b) <>
   (c) =
   (d) ==

4. Which operation always returns true when a is not less than b?
   (a) a != b
   (b) a < b
   (c) a >= b
   (d) a == b

5. Which operator returns true when two things are equal?
   (a) ==
   (b) <=>
   (c) !=
   (d) =

6. What does the == operator do?
   (a) creates (or updates) a new Boolean variable
   (b) gives a new value (but the variable must already exist)
   (c) compares two things for equality
   (d) creates a new variable

Concept: combining Boolean values with and, or, and not

7. What is the value of a?
   
   ```
   y = 4;
   x = 2;
   a = y < x || y % 2 == 0;
   ```
   
   (a) true
   (b) no value, an error occurs
   (c) false
8. What is the value of $a$?

\[
y = 3; \\
x = 4; \\
a = y < x \text{ || } y \% 2 == 0;
\]

(a) \text{false} \\
(b) no value, an error occurs \\
(c) \text{true}

9. What is the value of $a$?

\[
y = 3; \\
x = 2; \\
a = y < x \text{ || } !(y \% 2 == 0);
\]

(a) \text{true} \\
(b) no value, an error occurs \\
(c) \text{false}

10. What is the value of $a$?

\[
y = 1; \\
x = 3; \\
a = y < x \text{ && } !(y \% 2 == 0);
\]

(a) no value, an error occurs \\
(b) false \\
(c) true

11. What is the value of $a$?

\[
y = 3; \\
x = 2; \\
a = !(y < x \text{ || } y \% 2 == 0);
\]

(a) false \\
(b) no value, an error occurs \\
(c) true

12. What is the value of $a$?

\[
y = 2; \\
x = 3; \\
a = !(y > x \text{ && } y \% 2 == 0);
\]

(a) false \\
(b) no value, an error occurs \\
(c) true

13. What is the value of $a$?

\[
y = 2; \\
x = 3; \\
a = !(y > x \text{ && } y \% 2 == 0);
\]

(a) \text{false} \\
(b) \text{true} \\
(c) no value, an error occurs

14. What is the value of $a$?

\[
y = 2; \\
x = 3; \\
z = 0; \\
a = !(y > x \text{ / } z \text{ && } z \neq 0);
\]

(a) no value, an error occurs
15. What is the value of $a$?

```c
y = 2;
x = 3;
z = 0;
a = !(z != 0 && y > x / z);
```

(a) false
(b) no value, an error occurs
(c) true

**Concept:** short-circuiting

16. T or F: C will evaluate the expression $y \% 2 == 0$.

```c
y = 4;
x = 2;
a = y < x || y \% 2 == 0;
```

17. T or F: C will evaluate the expression $y \% 2 == 0$.

```c
y = 4;
x = 2;
a = !(y < x || y \% 2 == 0);
```

18. T or F: C will evaluate the expression $y \% 2 == 0$.

```c
y = 4;
x = 2;
a = x < y || y \% 2 == 0;
```

19. T or F: C will evaluate the expression $y \% 2 == 0$.

```c
y = 4;
x = 2;
a = !(x < y || y \% 2 == 0);
```

20. T or F: C will evaluate the expression $y \% 2 == 0$.

```c
y = 3;
x = 4;
a = y < x && y \% 2 == 0;
```

21. T or F: C will evaluate the expression $y \% 2 == 0$.

```c
y = 3;
x = 4;
a = !(y < x && y \% 2 == 0);
```

22. T or F: C will evaluate the expression $y \% 2 == 0$.

```c
y = 3
x = 4
a = x < y && y \% 2 == 0
```

23. T or F: C will evaluate the expression $y \% 2 == 0$.

```c
y = 3;
x = 4;
a = !(x < y && y \% 2 == 0);
```

**Concept:** if statements

24. Consider this code:

```c
if (P)
    printf("A\n");
else
    printf("B\n");
```
Mark the true statement.
(a) Either A or B will be printed, but not both
(b) It is possible that neither A nor B is printed
(c) It is possible that A and B can both be printed

25. Consider this code:
   ```c
   if (P)
       printf("A\n");
   if (Q)
       printf("B\n");
   ```
Mark the false statement.
(a) It is possible that neither A nor B is printed
(b) Only one of A and B can be printed
(c) It is possible that A and B can both be printed
(d) It is possible that only A is printed

26. Mark the true statement:
   (a) !!!b is the same as b
   (b) !!!b is the same as !(b)
   (c) !!!b is the same as !( !(b))

27. Mark the true statement:
   (a) !(a && b) is the same as !a || !b
   (b) !(a && b) is the same as !a && b
   (c) !(a && b) is the same as !a || b
   (d) !(a && b) is the same as !a && !b

28. Mark the true statement:
   (a) !(a || !b) is the same as !a && b
   (b) !(a || !b) is the same as !a && !b
   (c) !(a || !b) is the same as a || !b
   (d) !(a || !b) is the same as !a || !b

29. The variables a and b hold numbers. Mark the true statement:
   (a) !(a < b) is the same as a > b
   (b) !(a < b) is the same as a >= b
   (c) !(a < b) is the same as !a > !b
   (d) !(a < b) is the same as !a >= !b

30. The variables a and b hold numbers. Mark the false statement:
   (a) !(a == b) is the same as a != b
   (b) !(a == b) is the same as a > b && a < b
   (c) !(a == b) is the same as !(a <= b && a >= b)
   (d) !(a == b) is the same as a > b || a < b

31. Consider the following code:
   ```c
   if (P)
       printf("A\n");
   else if (Q)
       printf("B\n");
   else
       printf("C\n");
   ```
Mark the true statement.
(a) it is possible that C, but not B, is printed
(b) it is possible that A and B can both be printed
(c) it is possible that B and C can both be printed

(d) it is possible that A, B, and C can all be printed

(e) it is possible that no values are printed

32. Consider the following code:

```c
if (P)
    printf("A\n");
if (Q)
    printf("B\n");
else
    printf("C\n");
```

Mark the true statement.

(a) it is possible that A, B, and C can all be printed
(b) it is possible that A and B can both be printed
(c) it is possible that no values are printed
(d) it is possible that B and C can both be printed

33. Consider the following code:

```c
if (P)
    printf("A\n");
else if (Q)
    printf("B\n");
else
    printf("C\n");
```

Mark the true statement.

(a) one, two, or three values will be printed in total
(b) two or three values will be printed in total
(c) one or two values will be printed in total
(d) one value will be printed in total

34. Consider the following code:

```c
if (P)
    printf("A\n");
if (Q)
    printf("B\n");
else
    printf("C\n");
```

Mark the true statement.

(a) one or two values will be printed in total
(b) zero or two values will be printed in total
(c) two or three values will be printed in total
(d) one, two, or three values will be printed in total

35. Consider the following code:

```c
if (P)
    printf("A\n");
if (Q)
    printf("B\n");
if (R)
    printf("C\n");
```

Mark the true statement.

(a) zero, one, two, or three values will be printed in total
(b) one, two, or three values will be printed in total
(c) zero or three values will be printed in total
(d) one or two values will be printed in total
36. Consider the following code:

```c
if (P)
    printf("A\n");
else if (Q)
    printf("B\n");
else if (R)
    printf("C\n");
```

Mark the true statement.

(a) one or two values will be printed in total
(b) zero, one, two, or three values will be printed in total
(c) zero values or one value will be printed in total
(d) one, two, or three values will be printed in total

37. Consider the following code:

```c
if (P)
    printf("A\n");
else if (Q)
    printf("B\n");
else if (R)
    printf("C\n");
```

Mark the false statement.

(a) it is possible that C is printed if A and B are not
(b) exactly one value will be printed
(c) it is possible that no values will be printed
(d) it is possible that only one value will be printed

38. Consider the following code:

```c
if (P)
{
    if (Q)
        printf("A\n");
}
```

Mark the true statement.

(a) one value will be printed in total if P is true
(b) one value will be printed in total if P is false
(c) zero values or one value will be printed in total
(d) no values will be printed in total if P is true

39. Consider the following code:

```c
if (P)
{
    if (Q)
        printf("A\n");
    else
        printf("B\n");
}
```

Mark the true statement.

(a) one or two values will be printed in total
(b) one value will be printed in total
(c) zero values or one value will be printed in total
(d) zero or two values will be printed in total

40. What is the possible output of this code fragment after the input statement is performed?

```c
x = readInt(stdin);
if (x % 2 == 0)
    printf("even");
printf("done");
```
41. What is the possible output of the if statement?

```c
x = readInt(stdin);
if (x % 2 == 0)
    printf("even");
else
    printf("odd");
```

(a) both even and odd
(b) even or odd, but not both
(c) even
(d) odd

42. What is not a possible output of the if statements?

```c
x = readInt(stdin);
if (x < 100)
    printf("smaller");
if (x < 1000)
    printf("bigger");
```

(a) nothing is printed
(b) smaller
(c) bigger
(d) both smaller and bigger are printed (in that order)