Instructions: This homework assignment focuses primarily on some of the basic syntax and semantics of C++. The answers to the following questions can be determined from Topics 7 and 8 of the lecture notes and Chapters 5 and 7.1 of the text.

After you have analyzed the questions and decided what answers you believe are correct, you may find it useful to write some short programs to test your logic. Submit your answers to the Curator collection point `HWQuiz4`.

For questions 1 through 4, consider executing the code fragment:

```
bool A, B, C, D;
// code that assigns values to A, B, C, and D
if (A && B)
   if (!C || !D)
      cout << "one" << endl;
   else if (D)
      cout << "two" << endl;
   else
      cout << "three" << endl;
else if (C != D)
      cout << "four" << endl;
else if (C)
      cout << "five" << endl;
else
      cout << "six" << endl;
```

1. What of the following sets of values for A, B, C, and D would cause the string "one" to be printed?

	A	B	C	D
1)	true	true	true	true
2)	true	true	true	false
3)	true	true	false	true
4)	true	true	false	false

5) All of the above
6) 1 and 2 only
7) 1, 2 and 3 only
8) 2 and 3 only
9) 2, 3 and 4 only
10) None of these

2. What of the following sets of values for A, B, C, and D would cause the string "four" to be printed?

	A	B	C	D
1)	true	false	true	true
2)	false	true	false	true
3)	false	true	true	false
4)	false	false	false	false

5) All of the above
6) 1 and 2 only
7) 1, 2 and 3 only
8) 2 and 3 only
9) 1, 3 and 4 only
10) None of these

3. No matter what values A, B, C and D have, the given code fragment will produce <u>some</u> output.

1) True 2) False 3) None of these

4. What of the following sets of values for A, B, C, and D would cause the string "three" to be printed?

	A	B	C	D
1)	true	true	true	true
2)	true	true	true	false
3)	true	true	false	true
4)	true	true	false	false

5) All of the above
6) 1 and 2 only
7) 1, 2 and 3 only
8) 2 and 3 only
9) 1, 3 and 4 only
10) None of these

---

5. Consider executing the following code fragment (assume `x` is an `int` variable):

```
if ( x <= 0 )
   cout << "One" << endl;
else if ( x <= 10 )
   cout << "Two" << endl;
else if ( x <= 20 )
   cout << "Three" << endl;
```

The string "Two" will be printed if and only if `x` satisfies the condition:

1) `x <= 10`
2) `0 < x and x < 10`
3) `0 < x and x <= 10`
4) `0 <= x and x < 10`
5) `0 <= x`
6) None of these

For questions 6 and 7, consider execution of the following C++ `switch` statement:

```
int Enter = 10;
cin >> Enter;

switch (Enter) {
case 1:  Enter = -4;
case 2:  Enter = -6;
case 4:  break;
case 6:  Enter = -8;
         break;
default: Enter = -1;
}
```

6. What would the value of `Enter` be after execution of this code if the value read for `Enter` were 4?

1) -4
2) -6
3) -8
4) -1
5) 10
6) None of these

7. What would the value of `Enter` be after execution of this code if the value read for `Enter` were 1?

1) -4
2) -6
3) -8
4) -1
5) 10
6) None of these

8. Consider the C++ expression: `!(x < 10 || x >= 14)`

Which of the following C++ expressions are equivalent to the one given above? Two expressions are equivalent if they will always have the same value, no matter what values the variables in them have.

1) `x < 10 && x >= 14`
2) `x > 10 && x <= 14`
3) `x >= 10 && x < 14`
4) `x >= 10 || x < 14`
5) `x > 10 || x <= 14`
6) None of them are equivalent.

---

9. Consider the C++ expression: `!(A && !B || C)`

Which of the following C++ expressions are equivalent to the one given above? Two expressions are equivalent if they will always have the same value, no matter what values the variables in them have.

1) `!A && B || !C`
2) `!A || B && !C`
3) `!A && !B || C`
4) `!A || !B && C`
5) None of them are equivalent.

---

10. What happens if the following C++ code is executed:

```
int x = 0, y = 14;

if ( x != 0 && y / x > 12 )     // Line 1
   cout << "case 1" << endl;    //      2
else if ( x != 0 )              //      3
   cout << "case 2" << endl;    //      4
else                            //      5
   cout << "case 3" << endl;    //      6
```

1) A run-time error will occur in line 1 when 14 is divided by 0.
2) The string "case 1" will be written.
3) The string "case 2" will be written.
4) The string "case 3" will be written.
5) None of these.