

Ray Mitchell, U99999999
MeanOldTeacher@MeanOldTeacher.com
C/C++ Programming I
Section 146359, Ray Mitchell
June 25, 2019
C1A1E0_Quiz.txt
Quiz Answers

1. D
2. E
3. B
4. B
5. D
6. A

C1A1E0 Explanations

In addition to the course book references cited below, these topics are also covered in the live lectures (in-class students) and the recorded lectures (online students).

1. **D** Note 1.5; One or more characters between double quotes form a string literal, except that to represent a backslash character or double-quote character inside a string literal that character must be preceded by a backslash.
2. **E** Note 1.5; One or (or sometimes) more characters between single quotes form a character literal, except that to represent a backslash character or a single-quote character inside a character literal that character must be preceded by a backslash.
3. **B** Note 1.5; `printf("\x49\x146\x155\x155\x71\x0021")` uses a sequence of 3 octal and 3 hexadecimal escape sequences to represent the values of the 6 ASCII characters *lfmmq!* Representing the values of characters numerically is an extremely bad practice since which characters these values represent is both cryptic and non-portable.
4. **B** Note 1.11; Only types **char**, **short**, and **int** are acceptable for `%d` in `printf`.
5. **D** Notes 1.15 and 1.16; `%c` in `scanf` does not skip leading whitespace. Precede it with `\n` to accomplish this task. `getchar` and `cin.get` do not skip leading whitespace. `cin >>` always skips leading whitespace unless explicitly changed with the `noskipws` manipulator.
6. **A** Note 1.13; Only type **int** is acceptable for `%d` in `scanf`.

```
1  //
2  // Ray Mitchell, U999999999
3  // MeanOldTeacher@MeanOldTeacher.com
4  // C/C++ Programming I
5  // Section 146359, Ray Mitchell
6  // June 25, 2019
7  // C1A1E1_main.cpp
8  // Windows 10 Professional
9  // Visual Studio 2019 Professional
10 //
11 // This file contains function main, which displays the most appropriate forms
12 // of various expressions.
13 //
14
15 #include <iostream>
16
17 //
18 // Function main displays the most appropriate forms of various expressions
19 // with respect to the value variable ax will contain after each expression
20 // is evaluated.
21 //
22 int main()
23 {
24     std::cout <<
25         "\"ax = ax + bx\" should be \"ax += bx\"\\n"
26         "\"ax = ax / -bx\" should be \"ax /= -bx\"\\n"
27         "\"ax = bx / ax\" should be \"ax = bx / ax\"\\n"
28         "\"ax = -1 * ax\" should be \"ax = -ax\"\\n"
29         "\"ax = -ax * ax\" should be \"ax *= -ax\"\\n"
30         "\"ax = -bx * ax\" should be \"ax *= -bx\"\\n"
31         "\"ax = bx - ax\" should be \"ax = bx - ax\"\\n"
32         "\"ax = 2 + ax\" should be \"ax += 2\"\\n"
33         "\"ax = 1 + ax\" should be \"++ax\" or \"ax++\"\\n"
34         "\"ax = ax - 37\" should be \"ax -= 37\"\\n"
35         "\"ax = ax - 1\" should be \"--ax\" or \"ax--\"\\n"
36         "\"ax *= -1\" should be \"ax = -ax\"\\n"
37         "\"ax /= -1\" should be \"ax = -ax\"\\n"
38         "\"ax = 0 - ax\" should be \"ax = -ax\"\\n";
39
40     return 0;
41 }
```

```
1  //
2  // Ray Mitchell, U99999999
3  // MeanOldTeacher@MeanOldTeacher.com
4  // C/C++ Programming I
5  // Section 146359, Ray Mitchell
6  // June 25, 2019
7  // C1A1E2_main.c
8  // Windows 10 Professional
9  // Visual Studio 2019 Professional
10 //
11 // This file contains function main, which evaluates and displays a
12 // polynomial.
13 //
14
15 #include <stdio.h>
16 #include <stdlib.h>
17
18 //
19 // Function main evaluates and displays the value of a polynomial that
20 // includes the squaring and cubing of a user input value.
21 //
22 int main(void)
23 {
24     // Prompt the user for input and read it.
25     double x;
26     printf("Enter a decimal value: ");
27     scanf("%lg", &x);
28
29     // Compute and display the value of a polynomial.
30     double square = x * x;
31     double result = 3 * square * x - 5 * square + 6;
32     printf("If x = %g the value of \"3x^3 - 5x^2 + 6\" is %g\n", x, result);
33
34     return EXIT_SUCCESS;
35 }
```