Mariya Eggensperger

Prof. Pam Wiese

C++ Programming I

Fall 2015

**1. Write a statement that would create the following output (including the " "):**

**"Hello, there!"**

#include <iostream>

using namespace std;

int main()

{

cout << "\"Hello world!\"";

return 0;

}

**2. What two data types would be appropriate to store the price of an item?**

A floating-point number (there are two types of floats that differ in their storage capacity: float 4 bytes and the double/long double 8 bytes) would be one appropriate way to store the price of an item. The float rather than the double / long double are more appropriate in that sense. Additionally, an integer would also be appropriate to store the price of an item. Those numbers with decimals are assigned a floating point variable and those without a decimal often utilize the integer variable.

**3. What data type would be appropriate to store the answer to a True/False question, assuming the values store are either T or F?**

The bool data type, or the Boolean, is used to represent logical data and is restricted to one of two values being True/False.

**4. What data type would be appropriate to store a person's middle initial?**

The char variable type is used to store single characters, including the letters of the alphabet (upper and lowercase) and is therefore the appropriate data type to store a person's middle initial.

**5. What data type would be appropriate to store a book title?**

A string data type would be appropriate to store a book title becuase it is a data class that represents text string. A string for a book title might look something like:

string bookTitle= "Pride and Prejudice"

**6. What data type would be appropriate to keep track of how many students enroll in a course?**

The int data type is most appropriate to keep track of how mnay students enroll in a course because the values are supported by the int data type are whole numbers mathematically referred to as integers. Integers are without decimal points.

**7. Write the code to declare an integer called "score" and initialize the variable to zero.**

#include <iostream>

using namespace std;

int main()

{

int score;

cout << "What is your score? ";

cin >> score;

cout << "Your score is " << score << "."; // Notice space after 'is.'

return 0;

}

**8. Write the code to declare a variable named "name" that will store a person's first name.**

#include <iostream>

using namespace std;

int main()

{

string firstName;

int score;

cout << "What is your name? ";

cin >> firstName;

cout << "What is your score? ";

cin >> score;

cout << "So, " << firstName << ", your score is " << score << ".";

return 0;

}

**9. Write the code to assign a value of "Buster" to the variable you created in #8.**

#include <iostream>

using namespace std;

int main()

{

string firstName= "Buster";

int score=0;

cout << "So, " << firstName << ", your score is " << score << ".";

return 0;

}

**10. Which of the following are valid identifiers for variables? If not valid, explain why not.**

a. quizScore: valid, mnemonic

b. ~~#ofCars~~: invalid, begins with a special character

c. ~~total people~~: invalid, blank spaces aren't allowed

d. value4: valid, followed by a digit (or letter)

e. jersey\_number: valid

f. ~~cartoon-character~~: invalid, contains a special character and a keyword (char)

g. ~~3points~~: invalid, begins with a number

11. Write a single line of code that declares two variables, firstName and lastName, as string variables.

#include <iostream>

using namespace std;

int main()

{

string firstName="Jane", lastName="Smith";

cout << "Hello there, Ms. " << firstName << "\n" << lastName << ".";

return 0;

}

**12. Write the code to declare 3 variables, month, day and year, using one declaration, but splitting the declaration onto 3 lines. Put a comment for each of the variables: for month - value between 1 - 12, for day - value between 1 - 31, for year - value between 1880 and 2015.**

#include <iostream>

using namespace std;

int main()

{

int varDay; // values between 1-31

string varMonth; // values between 1-12

int varYear; // values between 1880-2015

cout <<"What day is it? ";

cin >> varDay;

cout <<"What month is it? ";

cin >> varMonth;

cout << "What is the year? ";

cin >> varYear;

cout << "It is the, " << varDay << " of " << varMonth<< "\n" << varYear << ".";

return 0;

}

**13. Write the code to declare three integer variables, length, width, and area. Assign the value 10 to length and 5 to width.**

#include <iostream>

using namespace std;

int main()

{

int length=10;

int width=5;

int area; // This is the area of a rectanble. Measurements are in feet.

// Note: Final answer must be be in square inches.

area=length\*width;

cout << length << "\*" << width << "=" << area<< "ft^2" << endl;

return 0;

}

**14. Write a statement to compute the area by multiplying length times width.**

A= l\*W

**15. Write a statement to output the three values.**

cout << length << "\*" << width << "=" << area<< "ft^2" << endl; // Measurements are in feet.

**16. Write another statement to output the three values, but this time with text that indicates what each value stands for.**

Sample Output:

Length: 10

Width: 5

Area: 50

#include <iostream>

using namespace std;

int main()

{

int length=10;

int width=5;

int area= 50; // This is the area of a rectangle. Measurements are in feet.

// Note: Final answer must be be in square inches.

area=length\*width;

cout << length << "\*" << width << "=" << area<< "ft^2" << endl;

return 0;

}

**17. List and differentiate among the three types of errors that can occur in a program.**

There many types of errors that can occur in a program. The three primaryare:

Syntax errors: mistakes in programming code.

Compiler errors: events that cause the program to crash.

Logic errors: the program runs, but produces incorrect results.

Examples may include:

* Undeclared variables: the C++ compiler this error and sends the programmer an error message.
* Meaningless expressions due to premature vlaue assignment: whatever value originally to be used in the variable is applied when the expression is evaluated and thus, the expresion results in error
* Poor mathematical operations: Difficult to detect
* Mixing data types without understanding the effect of each
* Sloppy programming and forgeting to add << or >> for the cin/couts
* Storing data in the wrong declared data type.