

Concept review questions

This collection of concept questions includes one question per topic. It is meant to gage your strength of understanding of one aspect per topic, to help guide you as to what you need to study. The questions are very targeted towards each topic; however, midterm questions will almost certainly cover numerous topics per question.

1. Describe, in your own words, the steps in generating an executable program.
2. Define a statement and a block of statements.
3. Describe, in your own words, the purpose of escape sequences.
4. What is the difference between reserved identifiers and keywords?
5. Suppose you were dividing an unsigned integer m by an unsigned integer n . What would you have to do to ensure that result of the calculation moves to the closest integer, as opposed to rounding towards zero. You will have to add something from m , as the straight-forward calculation m/n will round towards zero. What happens if your calculation is something like $11/2$?

Bonus: What do you have to do if m and n are signed integers?

6. Explain in your own words why `std::cout << 1 << 2 << 3 << 4 << std::endl;` works as expected, as it is required that the left-hand operand is always a `std::cout` object.
7. In C++, a function can have many parameters, but only one return value. Suppose you wanted to create a programming language where a function could have many return values. How would you change the function declarations and definitions to accommodate this change?

Note: There is no “correct” solution to this question, there are only solutions that are reasonable and internally consistent.

8. Describe the purpose of documentary, functional, algorithm and explanatory comments.
9. In secondary school, you may have used different types of bracketing symbols to allow the reader to clearly visualize matching parenthesis. Is this valid in C++?

```
return {[(3.2*x + 4.7)*x + 7.2]*x + 5.2}*x;
```

10. Define a code *skeleton*.
11. Describe a decision making process you had today that involved more than two options, and where the criteria were vague. What decision did you make?
12. What is meant by two conditional statements being *complementary*?
13. Describe the format of and the behavior of a cascading conditional statement.

14. What are the return values of the following logical operations?

```
bool f1( bool a, bool b ) {  
    return (!(a || b)) == (!(a) && (!b));  
}
```

```
bool f1( bool a, bool b ) {  
    return (!(a && b)) == (!(a) || (!b));  
}
```

How would you describe the relationship between the two values that are being compared for equality?

15. What is the purpose of a function? Why not just write the code that does what you require at a particular point in time in your program? List three issues that result from simply implementing whatever functionality you require, as opposed to writing a function to perform that common operation.

16. Describe how a call stack allows a function to call itself with arguments other than those originally passed to the function.

17. If you execute

```
std::cout << "Enter some text: ";  
std::string input;  
std::cin >> input;
```

and the user enters the 12 characters `Hello world!` and presses Enter, what is the string stored in the local variable `input`?

18. Draw the flow charts for both a conditional statement in C++ and a repetitious statement in C++.

19. Write a flow chart for Programming Question 20.

20. In your own words, describe the structured programming theorem.

21. Write a flow chart for Programming Question 21.

22. Write a flow chart for Programming Question 22.

23. Write a flow chart for Programming Question 23.

24. What binary numbers come before and after `0b101101101111111111`? What hexadecimal numbers come before and after `0x2dbff`? What is the relationship between these two numbers, one in binary and the other in hexadecimal?

25. Why do negative numbers always go from -2^n up to -1 , but positive numbers only go from $2^n - 1$? Be sure to refer to the representation of negative numbers.