

Quiz 1

Solutions

Answers other than the below may be possible.

Multiple Choice.

- 0. d
- 1. b
- 2. d
- 3. c

True or False.

- 4. T
- 5. F
- 6. F
- 7. T

Sesame Street.

8.

Zero, ah ah ah...	000
One, ah ah ah...	001
Two, ah ah ah...	010
Three, ah ah ah...	011
Four, ah ah ah...	100
Five, ah ah ah...	101
Six, ah ah ah...	110
Seven, ah ah ah...	111

9.

decimal	binary	hexadecimal
40	101000	28

Shuttleboy.

```
10. <!DOCTYPE html
    PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
  <head><title>Quiz 1</title></head>
  <body>
    <table>
      <tr>
        <td>Date</td>
        <td>Departure Time</td>
        <td>Arrival Time</td>
      </tr>
      <?
        $fp = fopen(URL, "r");
        fgetcsv($fp);
        while ($row = fgetcsv($fp))
        {
          print("<tr>");
          print("<td>{$row[0]}</td>");
          print("<td>{$row[1]}</td>");
          print("<td>{$row[2]}</td>");
          print("</tr>");
        }
        fclose($fp);
      ?>
    </table>
  </body>
</html>
```

Sectioning.

```
11. http://section.cs50.net/section.php?name=Drew&comfort=less

12. function validate()
    {
      if (document.forms.section.name.value == "")
        return false;
      else
        return true;
    }
```

Back to C.

13.

<code>sizeof(char)</code>	1
<code>sizeof(char *)</code>	8
<code>sizeof(int)</code>	4
<code>sizeof(int *)</code>	8

14. During **preprocessing**, directives like

```
#include <cs50.h>
#include <stdio.h>
```

are effectively replaced with the contents of `cs50.h` and `stdio.h`, respectively, thereby supplying the compiler with prototypes for `GetString` and `printf`.

The preprocessed C code is then **compiled** (*i.e.*, translated) into assembly instructions, possibly with optimizations applied.

Those assembly instructions are then **assembled** (*i.e.*, translated again) into machine code and stored in an object (`.o`) file.

The object code for `main` is then **linked** (*i.e.*, combined) with that for `GetString` and `printf`, the result of which is an executable binary.

Rapid Fire.

15. Languages are like tools: you should use the right tool for the job. It's much easier to parse files and strings with PHP than with C; if you can afford to spend more cycles on some task, you might as well use PHP, since it will take less time (and fewer lines of code) to get the job done.
16. To obfuscate code means to make it less human-readable, as by shortening parameters' and variables' names, removing whitespace, or even encrypting it. The goal is to make it harder for others to read, understand, and/or reuse the code.
17. A primary key is a field that can be used to identify uniquely rows a database table.
18. Ajax is a technique whereby webpages can request additional data (*e.g.*, JSON) from a server via JavaScript without having to reload entirely.
19. An associative array is a data structure that allows programmers to associate values with keys, the latter of which can be numbers or even strings.

20. Little-endian machines store the least-significant byte of multibyte values at the lowest address in memory. For instance, with respect to an `int` like `0x0A0B0C0D`, if `0D` is at location `a` in memory, then `0C` would be at location `a+1`, `0B` would be at location `a+2`, and `0A` would be at location `a+3`.
21. An API is an interface (*e.g.*, a set of functions) written by programmers for other programmers can use in their own software. Generally, the latter need not know how the API is implemented underneath the hood; they need only know how to interact with it.

“Valgrind hates me.”

22. Alice has likely written to a 4-byte location in memory that does not belong to her program. For instance, she might have allocated an array, `a`, of size 10 but then written to location `a[10]`, in which case she should fix her code to not go past `a[9]`.
23. Dev has likely allocated 40 bytes of memory (as by allocating an array of 10 `ints`) but failed to deallocate them; he should take care to call `free` on whatever pointer was returned by `malloc`.

Er, not really.

24. XHTML is a markup language that lets you specify the structure and aesthetics of a page. It does not let you “program” because (for the most part) it does not allow you to implement logical flow control, which actual programming languages do.
25. Huffman coding promises lossless compression, whereby no actual information is lost as a result of compression. If any file’s size could be reduced to just one bit, it would be impossible to decompress it deterministically (*i.e.*, reliably), since that bit could map back to any file, and so information could be lost, thereby violating Huffman’s promise.

Bitwise.

26. This implementation works as it’s supposed to, as it returns `false` if and only if `n` has a least-significant (*i.e.*, right-most) bit of 1. Odd, not even, numbers have least-significant bit of 1, and so this implementation correctly returns `true` whenever `n` is even.
27. This implementation works as it’s supposed to (so long as `n` is greater than or equal to -2^{30}), as it shifts `n`’s bits leftward by one position each, the result of which is to double `n`’s value, since the value of each bit in an `int` is twice that of its right-hand neighbor.¹

¹ Actually, that’s not quite true. Because `double` is a type, you can’t use it as the name of a function, but noticing that subtlety wasn’t the intent of this question. Arguments either way were thus accepted.

Design Decisions.

28. MySQL should be used when you have a lot of data that you'd like to be able to search efficiently. MySQL can build indexes for fields that expedite searches. CSV files must be searched linearly.
29. JSON should be used when you'd like to associate non-numeric keys with values, so that you needn't worry about fields' relative ordering. JSON should also be used when you'd like to represent hierarchical structures. A CSV file, by contrast, do not allow for hierarchy, and you must know a priori which fields are in which columns (unless the CSV file's header row contains fields' names).
30. A hash table should be used when you have a sparse data set (*e.g.*, words that tend not to share long common prefixes), as a trie's nodes (each of which contains an array) would likely take up much more space.
31. POST should be chosen over GET when large amounts of data are to be submitted from client to server. Or when sensitive data (*e.g.*, passwords) need to be submitted.

Binary Trees.

32. A name like `find` is probably ideal, as this function searches a binary tree exhaustively, returning `true` if and only if `n` is present somewhere in the tree
33.

```
void unload(node *ptr)
{
    if (ptr == NULL)
        return;
    unload(ptr->left);
    unload(ptr->right);
    free(ptr);
}
```