

Computer Science Contest #1112-07 Key

December 03, 2011

- |       |       |
|-------|-------|
| 1) B  | 21) C |
| 2) E  | 22) E |
| 3) C  | 23) E |
| 4) B  | 24) C |
| 5) A  | 25) A |
| 6) E  | 26) E |
| 7) B  | 27) B |
| 8) B  | 28) D |
| 9) D  | 29) E |
| 10) C | 30) C |
| ■     | ■     |
| 11) D | 31) D |
| 12) C | 32) C |
| 13) E | 33) C |
| 14) E | 34) C |
| 15) C | 35) D |
| 16) A | 36) A |
| 17) A | 37) A |
| 18) A | 38) B |
| 19) B | 39) E |
| 20) C | 40) D |
| ■     | ■     |

**Note to Graders:**

- All provided code segments are intended to be syntactically correct, unless otherwise stated (e.g. error is an answer). **Ignore any typographical errors.**
- Any necessary Standard Java 2 Packages are assumed to have been imported as needed.
- Assume any undefined (undeclared) variables have been defined as used.

## Brief Explanations:

1. 12 in base 13 \* 102 in base 7 = 765 in base 10. 23331 in base 4 is the only valid conversion.
2.  $3 * 2 = 6 + 2 = 8$  multiplication takes place before addition
3.  $7.2 = 2.5 = 9.7$
4. C7AT#9 is returned from the call to toUpperCase()
5.  $01 * 14 / 3 - 10 = -6$  as  $14 / 3 = 4 * 1 = 4 - 10 = -6$
6. lastIndexOf starts looking at length() - 1 and moves toward 0, returning the location of the match once found
7.  $2 \bmod 1 = 0 + 1 \bmod 2 = 1$   $0 + 1 = 1$
8.  $25 / 4 = 6$  there are no decimals as both numbers are integers
9. 3 is subtracted each time through the loop as long as the variable is greater than -2
10. k and m refer to the same array - [0] is changed to 4 and then to 7 - adding [0] together twice = 14
11. 2.75 is the last value printed out by the loop
12. Create a truth table after negating the expression. Plug in the values and see what happens.
13. All of the references in the array refer to null - calling methods on null generates a runtime error
14. Set, Map, and List will all work as the type of the ArrayList.
15.  $x \geq y * 7$  is the answer as none of the other options allow for x to be greater than or equal to
16. & happens 1<sup>st</sup>, then ^, and then | - 31 is the answer
17. Object does not implement the Comparable interface.
18. [12.5, 65.8, 16.6, 23.3, 3.6, 7.3] is the stack once all items are added
19. 3 pops would remove 7.3, 3.6, and 23.3 - peek would return, but not remove 16.6 - pop removes and returns 16.6
20. Stack is the only class that implements the List interface
21. we is shorter than web and it is first in the compareTo order so the method returns -1
22. The code generates a runtime error as 0x cannot be passed in to the constructor for Long.
23. Any of the boolean methods would work as the delimiter setting negates all punctuation marks.
24. 4.875 is average of all numbers in the string once +, -, and . are removed from the equation.
25. PriorityQueues in Java are implemented using min-heaps. The smallest value is removed.
26. PriorityQueues in Java are implemented using min-heaps. The smallest value is removed.
27. get(2) returns andrew as andrew was the only value associated with the key 2
28. get(7) returns jason as that was the last value associated with the key 7
29. There are 7 unique key, value pairs in the map. Maps do not store duplicate keys.
30. The class variable for CooDoo is used in the println.
31. TreeSet organizes the item by natural order using the compareTo from the Comparable interface.
32.  $i < n$  is the only condition that would allow the loop to run n-1 times
33. Method noob is collection all of the factors of n
34.  $O(n)$  is the appropriate runtime for method noob
35. 5 3 1 are the factors of 15
36.  $>>>$  divides by 2 - divide by 2 happens 3 times -  $32 - 16 - 8 - 4$
37. The exception needs to be instantiated in order to be thrown.
38. The output is true and then false as the first string ends with a 0 and the second string ends with a 1.
39. Method yo() is checking to see if the strings end with a 0.
40. D is the proper choice to check for groups of 1s.