

Computer Science Contest #1213-03 Key

October 27, 2012

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

**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.  $1101_3 = 37$ ,  $11_3 = 10$ ,  $1000_3 = 37 - 10 = 27$
2. int divided by double, uses double math
3.  $(15 \& 7) + 3 = 10$
4. I will be 10,12, and 14 when added: (i is incremented twice per loop)
5. "^" matches the beginning of the string, it would have to be escaped to literally mean the "^" symbol
6.  $a[a[a[0]/3]] \rightarrow a[a[11/3]] \rightarrow a[a[3]] \rightarrow a[2] \rightarrow 5$
7. "b2 || num++ > 0" short circuits, "b1 || num++ >= 1" doesn't, num is incremented a total to 2 times giving the value 3
8. longs cannot be used for the value in a switch-case
9. operator precedence  $5 + ((3\%4)*2) = 11$
10. I is correct, II is allowed by will give a warning, III is not allowed because primitive values cannot be generic parameters in Java
11. Math.round(double) returns a long
12. Printf can use a String as the format specifier.
13. It takes 2 \\ to make 1 \. \\n simply prints out \n.
14. matrix is created and looped through, summing values
15. simple nested loops except the "count+=j" part of the for loop
16. Strings are immutable, replaceAll's return value is ignored, the cool regex expression is irrelevant
17. operator precedence: and, xor, or,  $2^{(3\&5)} = 3$
18. operator precedence  $true | (false \wedge true) = true$
19. Generic parameter of the ArrayList is Long, but ints are attempted to be added. Java could box the int to an Integer, but it won't convert the Integer to the Long because they are siblings in the class hierarchy
20. assignment statement's expression value is the value that was assigned
21. "^" matches the beginning of the string, it would have to be escaped to literally mean the "^" symbol
22.  $rec(20) \rightarrow rec(10)+20 \rightarrow rec(5)+10+20 \rightarrow ""+10+20 = "1020"$
23.  $rec(8) \rightarrow rec(4)+8 \rightarrow rec(2)+4+8 \rightarrow rec(1)+2+4+8 \rightarrow ""+2+4+8 \rightarrow "248"$
24. The "a" and "b" references are of type "A", so both refer to "A"s static variables
25. The "a" and "b" references are of type "A", so both refer to "A"s public instance variables
26. In the class "A" the method "mnum" gets the value of the instance variable "num" in "A". In "B" however, "mnum" is overridden. "num" within the context of class "B" refers to "B"s "num", which shadows "A"s "num".
27. "i" increments normally, "j" increases exponentially
28. Outer loop is N, inner loop is logN because "j" increases exponentially
29.  $num(3) \rightarrow num(2)+num(1)+2 \rightarrow num(1)+num(0)+1+num(1)+2 \rightarrow 1+1+1+1+2 \rightarrow 5$
30.  $num(5) \rightarrow num(4)+num(3)+4 \rightarrow num(3)+num(2)+3+num(3)+4 \rightarrow$   
 $num(2)+num(1)+2+num(2)+3+num(3)+4 \rightarrow$   
 $num(1)+num(0)+1+num(1)+2+num(2)+3+num(3)+4 \rightarrow$   
 $1+1+1+1+2+num(1)+num(0)+1+3+num(3)+4 \rightarrow$   
 $1+1+1+1+2+1+1+1+3+num(2)+num(1)+2+4 \rightarrow$   
 $1+1+1+1+2+1+1+1+3+num(1)+num(0)+1+num(1)+2+4 \rightarrow$   
 $1+1+1+1+2+1+1+1+3+1+1+1+1+2+4 \rightarrow 22$
31.  $(15 \& 7) + 3 = 10$
32. Implements a binary tree. The array "num" is in the format {<value>, <index of left child>, <index of right child>, <value>, <index of left child>, <index of right child>, ...}. The method "thing" does a breadth-first search of the tree and counts the number of nodes it went

- through to find the value. The queue will be [0] -> [3,6] -> [6,0,12] -> [0,12,6,3] -> [12,6,3,3,6], the top index is 12, num[12] == 6, done
33. 3 is not a value of a node in the tree so it will never find it, and because there are loops in the tree, it will create an infinite loop
  34. It's obviously a binary tree. At the very least it's not a binary search tree because the elements are not ordered properly
  35. Elements are added to "a1" in reverse order that they were added to the stack (because it's a stack)
  36. A TreeSet will order elements in natural order
  37. The respective items from the answer of question 35 will be associated with the values from the answer of 36. A TreeMap will print based on the natural order of the keys
  38. the int pi is the current position in the string d, the string p is a set of instructions that are executed with the loop and the switch-case
  39. The regular expression ".i" matches any character followed by an "i"
  40. The regular expression "[.i]" matches a period or an "i"