Computer Science Answer Key
UIL Invitational A 2014

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

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. $11110_2 + 11011_2 = 30_{10} + 27_{10} = 57_{10} = 71_8 = 39_{16} = 111001_2$
2. b = 19 + 13 = 32, c = 13
3. Integer object array cannot be initialized with a double
4. j starts at 5, outputs from 4 down to 1, and stops at 1
5. character at position 4 is k
6. list2 is an alias for list 1, so any changes made by one are changes made to the other
7. p^q is p xor q, which requires opposites in order to be true. Since both are true, the result for p is false.
8. This string switch statement matches at "sweet", and outputs both "yum" and "yom" since the break is only after "yom"
9. The maximum of 5.2 and 3.1 is 5.2
10. The lengths of each row of this uneven grid are 3, 4, and 2
11. getNumStrings is an accessor method with a heading of public int since it returns an integer
12. and requires no parameter, so has empty ( ) with NO semicolon!
13. and simply returns the numStrings instance field value
14. Since & evaluates first in bitwise order, 15&7 results in 7, and then 25 | 7 results in 31.
15. This is a loop that calculates the log base 2 of 100…e increments by 1, but f doubles each time and passes 100 at the 7th iteration
16. The parameter in a method call is called the actual parameter
17. Since this is a chain if else, only one value is output for each call, according to the logic of the if statements. 6 produces 30, 9 produces 1, and 7 produces 7.
18. Both the (8) and (8,13) substring calls produce the word "Probe" from this string since the P is at position 8 and 13 is the length of the string, one step PAST the end of the substring desired.
19. This expression follows the order of operations and integer divides 12 and 5 to get 2, then adds 2.5 to get 4.5
20. The Boolean expression is p or q and q, which when simplified just becomes p (Law of Absorption) and therefore each output digit matches the p digit of the term.
21. Both I and II options correctly (but in different ways) isolate the 6. Option III isolates the 9.
22. 2PI is just a full circle..360 degrees.
23. Any integer left shifted 32 spots (the bit size of the integer data type), will simply return to its original value. Essentially it is a Left Circle back to the original number.
24. The list is indeed empty to start with, and after all the action, 3.1 is the sole surviving element in this list at position 0.
25. See the recursive trace on the right for the solution to this problem.
26. The "[elt]+" pattern splits at any sequence of the letters e, l and t, which produces the array ["Fr","Fa", "inTomP","y"] in this instance, producing "FryFa" for this output.

27. This ternary operation results in false, since 100%3 is not 0, therefore the resulting string is the one following the :, which is "bad", **very "bad"**!!!
28. The first different characters in these two strings are 'a' and 'i', which produces -8 since 'a' is 8 places before 'i'.
29. Since the value 10 can only map to one value, the "ten" is replaced with "sepuluh", another word for 10. Look it up!
30. Another digital electronics question! Don't you just love 'em? Just learn the basic symbols and this will become very easy for you. The bullet shape is the AND, and the arrow is OR. This is simply A and B or C.
31. DeMorgan's law is applied to the **not(a or b)**, resulting in **not a and not b**, which when "anded" with another **not a** simply becomes **not a and not b**.
32. Binary search is easy. Find the middle, and if doesn't match, go left or right and find the middle again. Repeat this process until the middle is the one you want. Then count the "middles" and that's how many steps you took to find it.
33. This structure is most certainly valid. Any class inheriting an abstract class is required to implement any abstract method in that class, so class B is REQUIRED to implement method one() from class A. Anything else is optional.
34. This is just simply 4 from method one times 5 from method two times the 2 from the variable x, for a result of 40.
35. All of these implementations of class B are valid. Look them over carefully.
36. Since p.next pointer references the second node of the list, and the data for that node is 2, the resulting output is 2.
37. In this TreeSet process, 4 is added twice, but since there are no duplicates in sets, only remains once. The 6 is removed, leaving only the 4, 5, and 7, so the size is 3 and 6 is not in the list.
38. Data input is a classic use of the try catch block. Since 3.14 is a mismatch for integers, the exception is thrown by the try block and caught by the catch block, resulting in the "Bad data. " output. The finally block ALWAYS occurs, no matter what.
39. In the heapify process of a min heap, the process always starts at the bottom right of the tree, working left and upwards, switching any parent and child values that are not in correct min heap order. The first such occurrence here is the 6 and 3. Next will be the 1 and 2, and so on.
40. In this queue push and pop sequence, the 3,5, and 4 are pushed, then the 3 is popped, push the 7, pop the 5, pop the 4, then push the 9. The 4 was the last value popped, and the 7 sits at the front of the queue, waiting to be popped next.