

# Computer Science Answer Key

## UIL District 1 2015

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

### 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.

## Explanations:

1.  $202_{10} + 10000101_2 = 202_{10} + 133_{10} = 335_{10} = 517_8 = 14F_{16} = 101001111_2$
2.  $17 + 7 - 1 / 5 = 17 + 7 - 0 = 17 + 7 = 24$
3. "A" + 10 + 50 ==> "A10" + 50 ==> "A1050"
4. Starting backwards from position 10 in the string, the last index of "abc" is 9.
5. Since p is true, NOT p is false, which makes the AND result false, regardless of q's value.
6. The cube root of 64 is 4.0.
7. The mod operator always results in a sign the same as the first operand, regardless of the sign of the second operand. Therefore, -55 mod -3.5 is the same as -55 mod 3.5, which is -2.5.
8. The values 10, 11, and 19 all make the OR conditional statement false, which when NOTed, becomes true, thus resulting in the "yes" output.
9. One star will be concatenated to the string s for each of the x values 4, 6, 8, 10, and 12. At 14, the while statement becomes false since 14 % 7 is equal to zero, resulting in an output of 5 stars.
10. The five values in the array are 0, 49, 50, 3, and 4, which sum to 106.
11. The Scanner class "lives" in the java.util package, which is the only one essential to this code for keyboard input. The java.io package must be included when using the File class for file input.
12. The sequence of values for y and x is: 1 100, 1 99, 2 97, 3 94, 4 90, 5 85, 6 79, 7 72, 8 64, 9 55, 10 55
13. The bitwise XOR operator (^) is on line 9 of the chart, followed by the logical OR operator (||) on line 12, and the ternary operators (? :) on line 13.
14. The Character data type uses 16 bits of storage.
15. The contains method does not work with the primitive array pList, thus the error. If it did, the answer would be **true false**, since pList does contain the value 30, but aList does not since the value 30 was divided by 10 first, then added to aList.
- 16.

$$\begin{array}{ccccc} A & B & \bar{A} & \bar{A} + B & \overline{\bar{A} + B} \end{array}$$

0	0	1	1	0
0	1	1	1	0
1	0	0	0	1
1	1	0	1	0

The truth table above shows three false results in the final column.

$\overline{\bar{A} + B}$  simplifies to  $A * \bar{B}$  using DeMorgan's law and the Double Negative Rule, showing (1,0) as the only true result.

17.  $20 + -15 / 5 ==> 20 + -3.0 ==> 17.0$
18. The value of each matrix element is the sum of the row and column index, which in the case of grid[3][2] is 3+2, which is 5.
19. Using the two's complement short-cut conversion process (see either 2015 Invitational Test for a complete explanation), 11001010 converts back to 00110110, which is the value 54, hence the original bit string is -54.
20. This is the classic insertion sort (descending order), where one by one, from the second element to the end of the list, the best place is found for each element. In the first iteration, nothing changed, since the 2 was already in the best slot at that time. In the second iteration, the 4 was swapped with the 2. On the third iteration, the 7 was moved towards the front two slots, moving the 4 and 2 to the right, resulting in the order 8 7 4 2 6.
21. This is the insertion sort, as described above.
22. In line 5, **temp > list[i - 1]** (descending order) must be changed to **temp < list[i - 1]**, which will make sort in ascending order.
23. The order of magnitude for the average and worst cases of the insertion sort is  $O(N^2)$ , with  $O(N)$  for the best case.
24. This process outputs each letter of the original string the number of times corresponding to its position in the alphabet, i.e., the E is output five times, the G seven times, etc.
25. The angle whose cosine is 0.5 measures 60 degrees
26. The Arrays.toString method outputs any char, double, int, long, or Object array in this form: [1.2, 3.45, 6.09, 5.4]
27. Since this is a method that is called in an output statement, it is a return method, which means <thing1> is the type data being returned, int, and <thing2> is the actual return statement, requiring the term return.
28. Since 9 mod 4 is not zero, 9 is the value returned. For 8, whose mod 4 value is 0, 8 \* 5, or 40 is returned.
29. The three instance fields are: private String type; private int numRooms; private boolean sturdy;
30. With block comments hiding both constructors, the compiler assigns default values to the three instance fields, which are then output using the toString method in the format shown, with null, zero, and false as the default values.
31. With both constructors uncommented, the provided values are shown in the output from the toString method.
32. The base 5 equivalent of 12 is 22, which essentially means 5 goes into 12 2 times, with 2 remainder.
33. The four popped values in sequence are 4, 6, 5, and 3, which sum to 18.
34. Preorder traversal starts at the root of the tree (top node), and "touches left" each node along all of the branches of the tree from left to right.
35. EABEC is not a simple path because it revisits a node, and DCBF is not a path at all since there is no path from B to F, only from F to B.
36. Since XOR has a lower precedence than AND, parentheses must surround the XOR expression since it comes first in the diagram.
37. **I N T E ^ / + L -** is the correct postfix equivalent for this expression.
38. The 5 possible values generated by this code range from 33 through 37, therefore 38 is not possible.

39.  $A \oplus B$   
Rationale:

$\overline{A * B + A + B}$  becomes  $(\bar{A} + \bar{B}) * (A + B)$  - DeMorgan's law on the \* and the +, and then double negative, which then becomes  $\bar{A} * A + \bar{A} * B + \bar{B} * A + \bar{B} * B$  using FOIL. The complement law further simplifies it to  $\bar{A} * B + A * \bar{B}$ , which is the definition for  $A \oplus B$

2015 DI
$f(5) = f(2) + 1 = 3 + 1 = 4$
$f(2) = f(-1) + 1 = 2 + 1 = 3$
$f(-1) = 2$

- 40.