★ANSWER KEY – CONFIDENTIAL★

UIL COMPUTER SCIENCE – 2021 INVITATIONAL A

Questions (+6 points for each correct answer, -2 points for each incorrect answer)

1) D	11) E	21) B	31) E
2) C	12) D	22) A	32) B
3) C	13) D	23) E	33) C
4) E	14) E	24) B	34) A
5) A	15) A	25) D	35) D
6) B	16) C	26) C	36) A
7) A	17) B	27) E	37) E
8) C	18) E	28) C	38) B
9) D	19) C	29) A	*39) O(n log n)
10) B	20) D	30) E	*40) 10011000

** See "Explanation" section below for alternate, acceptable answers.*

Note: Correct responses are based on **Java SE Development Kit 14 (JDK 14)** from Sun Microsystems, Inc. All provided code segments are intended to be syntactically correct, unless otherwise stated (e.g., "error" is an answer choice) and any necessary Java SE 14 Standard Packages have been imported. Ignore any typographical errors and assume any undefined variables are defined as used.

Explanations:

1. D

128	64	32	16	8	4	2	1
1	1	1	0	0	1	1	1

128 + 64 + 32 + 4 + 2 +1 = 231

2. C

9 / 4 * 3 + 7 – 2 = 2 * 3 + 7 – 2 = 6 + 7 – 2 = 13 – 2 = 11

3. C

The print methods leave the cursor on the same line where the text is printed. There are no spaces in the string literals so none are printed.

4. E

indexOf('a') returns the index value of the first occurrence 'a' in the string.

5. A

^ (XOR) will evaluate as true if one or the other but not both operands are true.
true ^ true ^ true =
false ^ true =
true

6. B

ceil(1.01) → 2.0, floor(-1.01) → -2.0, 2.0 +-2.0 = 0.0

7. A

8 *1.25 + 9 / 1.5 = 10.0 + 9 / 1.5 = 10.0 + 6.0 = 16.0

8. C

5 * 6 >= 13 is true
14 – 1 == 13 is true
5 + 6 – 13 > 14 is false
First output statement is not executed.
x + y + z = 5 + 6 + 14 = 25
Indentation does not create a block statement.

9. D

x	Output
0	
1	*
2	**
3	***
4	****
5	*****
6	******
7	*******

10. B

index	**0**	**1**	**2**	**3**	**4**	**5**	**6**
int[] list = …	0	4	2	6	3	5	1
list[1] = …	0	5	2	6	3	5	1
list[list[3]] = …	0	5	2	6	3	5	7

11. E

The scanner object should be closed instead of the file object.

12. D

g is added to h with each iteration of the loop.

h	**g**
1	2
3	3
6	4
10	5
15	6
21	7
28	8
36	9
45	10

13. D

++5 + 6 * 4 >> 1 =
6 + 6 * 4 >> 1 =
6 + 24 >> 1 =
30 >> 1 = *(this is the same as 30 / 2)*
15

14. E

byte -128 to 127
short -32768 to 32767
int -2147483648 to 2147483647

15. A

get returns the element at that index number.
remove removes the argument or returns false if it is not present.
indexOf returns the index number of the argument.

16. C

Constructors have the same name as the class and no return type.

<table>
<tr><td>17.</td><td>B</td><td>The default constructor sets i, d and s to 8, 0.5 and “science”. The first argument in the call to the constructor to instantiate obj2 is a call to obj1.d() which returns the string “nce” whose length is 3. So, the fields i, d and s in obj2 are assigned 3, 0.75 and “computer”. The call to obj2.d() returns “mputer”.</td></tr>
<tr><td>18.</td><td>E</td><td>obj3.d will not compile. d is a private field.</td></tr>
<tr><td>19.</td><td>C</td><td>x and y are two different list that contain the same elements. Therefore, x == y is false. The equals method is inherited from the abstract class AbstractList<E> which implements equals as follows: Compares the specified object with this list for equality. Returns true if and only if the specified object is also a list, both lists have the same size, and all corresponding pairs of elements in the two lists are equal.</td></tr>
<tr><td>20.</td><td>D</td><td>Cast first then add. Result is a double.</td></tr>
<tr><td>21.</td><td>B</td><td>values() returns an array containing each of the constants in the enumerated type.
name() returns the name of the constant.
Pets.CAT references the specific constant CAT.</td></tr>
<tr><td>22.</td><td>A</td><td>Letters are placed in the 2D array in row major order.</td></tr>
<tr><td>23.</td><td>E</td><td>5 + -3 + 65 + 4.25 =
2 + 65 + 4.25 =
67 + 4.25 =
71.25</td></tr>
<tr><td>24.</td><td>B</td><td>p = 6.3 += 12.85 %= 3 → p = 6.3 += 0.85 → p = 7.15 Compound assignment is done from right to left.</td></tr>
<tr><td>25.</td><td>D</td><td>Shift the bits two places to the right in the left operand to get 00101100_2 which is 44_{10}. The print method converts binary to decimal.</td></tr>
<tr><td>26.</td><td>C</td><td><table><tr><td>x</td><td>y</td></tr><tr><td></td><td>5</td></tr><tr><td>0</td><td>5</td></tr><tr><td>1</td><td>6</td></tr><tr><td>2</td><td>3</td></tr><tr><td>3</td><td>6</td></tr><tr><td>4</td><td>6</td></tr><tr><td>5</td><td>6</td></tr><tr><td>6</td><td>4</td></tr><tr><td>7</td><td>28</td></tr><tr><td>8</td><td>224</td></tr><tr><td>9</td><td>2016</td></tr></table></td></tr>
<tr><td>27.</td><td>E</td><td>A functional interface may only contain one abstract method but may also include default methods.</td></tr>
<tr><td>28.</td><td>C</td><td>table is an ArrayList of String arrays. The enhanced for loop gets each array in list and adds it to the end of the table as a row.</td></tr>
<tr><td>29.</td><td>A</td><td>Both the outer and inner loops execute 11 times. 11 * 11 = 121</td></tr>
<tr><td>30.</td><td>E</td><td>The second argument of the split method restricts the size of s to two elements.
print(s[2]) throws an indexOutOfBounds exception.</td></tr>
<tr><td>31.</td><td>E</td><td>If the string passed to method has an even length each half of the string is processed in a right to left fashion. If the string passed to the method has odd number of characters the middle character is printed then the remaining halves are processed from left to right. If the string passed to method has one character, that character is printed.</td></tr>
<tr><td>32.</td><td>B</td><td>A and D are selection sorts. C is a bubble sort.</td></tr>
<tr><td>33.</td><td>C</td><td><table><tr><td>Original Stack</td><td>add(1,”D”)</td><td>push “E”</td><td>add(0,pop())</td></tr><tr><td>C</td><td>C</td><td>E</td><td>C</td></tr><tr><td>B</td><td>B</td><td>C</td><td>B</td></tr><tr><td>A</td><td>D</td><td>B</td><td>D</td></tr><tr><td></td><td>A</td><td>D</td><td>A</td></tr><tr><td></td><td></td><td>A</td><td>E</td></tr></table></td></tr>
<tr><td>34.</td><td>A</td><td>Equivalent to NOT false OR A AND B. NOT false is true. true OR anything is always true.</td></tr>
<tr><td>35.</td><td>D</td><td>Arrays.sort(a, 3, 7) sorts the portion of the array a from index 3 to 6 in ascending order.
Arrays.copyOfRange(a, a[2], a[5]) is equivalent to copyOfRange(a,1, 8) which returns {7,1,4,6,8,9,2}</td></tr>
<tr><td>36.</td><td>A</td><td>Counts any character that is not a lower case letter.</td></tr>
</table>

37.	E	Apply & then |. 18 | 3 & 5 = 10010 | 011 & 101 = 10010 | 001 = 10011 = 19
38.	B	A class declared as an abstract class cannot be instantiated but an abstract class can have a constructor. Each abstract class must have a concrete subclass which will implement the abstract methods of that abstract class. Creating an object of any subclass invokes all the constructors in the corresponding inheritance tree. The same case applies to abstract classes. Even though an object of an abstract class cannot be instantiated, when an object of a class which is concrete and is a subclass of an abstract class is instantiated, the constructor of the abstract class is automatically invoked. Therefore, there can be a constructor in abstract class.
39.	O(n log n)	The outer loop executes n times and the inner loop executes $\log_2 n$ times. Multiply and ignore the base to get n * log n. Must be written using Big-O notation.
40.	10011000	Convert 104 to binary → 01101000 Flip all the bits → 10010111 Add one → 10011000