## **UIL COMPUTER SCIENCE WRITTEN TEST**

# 2025 REGION

## **APRIL 2025**

#### General Directions (Please read carefully!)

- 1. DO NOT OPEN THE EXAM UNTIL TOLD TO DO SO.
- 2. There are 40 questions on this contest exam. You will have 45 minutes to complete this contest.
- 3. All answers must be legibly written on the answer sheet provided. Indicate your answers in the appropriate blanks provided on the answer sheet. Clean erasures are necessary for accurate grading.
- 4. You may write on the test packet or any additional scratch paper provided by the contest director, but NOT on the answer sheet, which is reserved for answers only.
- 5. All questions have ONE and only ONE correct answer. There is a 2-point penalty for all incorrect answers.
- 6. Tests may not be turned in until 45 minutes have elapsed. If you finish the test before the end of the allotted time, remain at your seat and retain your test until told to do otherwise. You may use this time to check your answers.
- 7. If you are in the process of actually writing an answer when the signal to stop is given, you may finish writing that answer.
- 8. All provided code segments are intended to be syntactically correct, unless otherwise stated. You may also assume that any undefined variables are defined as used.
- 9. A reference to many commonly used Java classes is provided with the test, and you may use this reference sheet during the contest. AFTER THE CONTEST BEGINS, you may detach the reference sheet from the test booklet if you wish.
- 10. Assume that any necessary import statements for standard Java SE packages and classes (e.g., java.util, System, etc.) are included in any programs or code segments that refer to methods from these classes and packages.
- 11. NO CALCULATORS of any kind may be used during this contest.

#### Scoring

- 1. Correct answers will receive 6 points.
- 2. Incorrect answers will lose 2 points.
- 3. Unanswered questions will neither receive nor lose any points.
- 4. In the event of a tie, the student with the highest percentage of attempted questions correct shall win the tie.

#### STANDARD CLASSES AND INTERFACES – SUPPLEMENTAL REFERENCE

package java.lang class Object boolean equals (Object anotherObject) String toString() int hashCode() interface Comparable<T> int compareTo(T anotherObject) Returns a value < 0 if this is less than anotherObject. Returns a value = 0 if this is equal to anotherObject. Returns a value > 0 if this is greater than anotherObject. class Integer implements Comparable<Integer> Integer(int value) int intValue() boolean equals(Object anotherObject) String toString() String toString(int i, int radix) int compareTo(Integer anotherInteger) static int parseInt(String s) class Double implements Comparable<Double> Double (double value) double **doubleValue**() boolean equals (Object anotherObject) String toString() int compareTo (Double anotherDouble) static double parseDouble(String s) class String implements Comparable<String> int compareTo(String anotherString) boolean equals (Object anotherObject) int **length**() String **substring**(int begin) Returns substring (begin, length()). String substring (int begin, int end) Returns the substring from index begin through index (end - 1). int indexOf(String str) Returns the index within this string of the first occurrence of str. Returns -1 if str is not found. int indexOf(String str, int fromIndex) Returns the index within this string of the first occurrence of str, starting the search at fromIndex. Returns -1 if str is not found. int indexOf(int ch) int indexOf(int ch, int fromIndex) char charAt (int index) String toLowerCase() String toUpperCase() String[] split (String regex) boolean matches (String regex) String replaceAll(String regex, String str) class Character static boolean isDigit(char ch) static boolean **isLetter**(char ch) static boolean isLetterOrDigit (char ch) static boolean isLowerCase (char ch) static boolean isUpperCase (char ch) static char toUpperCase(char ch) static char toLowerCase (char ch) class Math static int **abs**(int a) static double **abs**(double a) static double pow(double base, double exponent) static double sqrt (double a) static double ceil (double a) static double floor(double a) static double **min**(double a, double b) static double max(double a, double b) static int **min**(int a, int b) static int **max**(int a, int b) static long round(double a) static double random() Returns a double greater than or equal to 0.0 and less than 1.0.

package java.util interface List<E> class ArrayList<E> implements List<E> boolean **add**(E item) int **size**() Iterator<E> iterator() ListIterator<E> listIterator() E get(int index) E set (int index, E item) void add(int index, E item) E **remove** (int index) class LinkedList<E> implements List<E>, Queue<E> void addFirst (E item) void addLast (E item) E getFirst() E getLast() E removeFirst() E removeLast() class Stack<E> boolean isEmptv() E peek() E pop() E push (E item) interface Queue<E> class PriorityQueue<E> boolean add (E item) boolean isEmpty() E peek() E remove() interface Set<E> class HashSet<E> implements Set<E> class TreeSet<E> implements Set<E> boolean **add**(E item) boolean contains (Object item) boolean **remove**(Object item) int size() Iterator<E> iterator() boolean addAll (Collection<? extends E> c) boolean removeAll(Collection<?> c) boolean retainAll(Collection<?> c) interface Map<K,V> class HashMap<K,V> implements Map<K,V> class TreeMap<K,V> implements Map<K,V> Object put (K key, V value) V get (Object key) boolean containsKey (Object key) int **size**() Set<K> keySet() Set<Map.Entry<K, V>> entrySet() interface Iterator<E> boolean **hasNext**() E next() void remove() interface ListIterator<E> extends Iterator<E> void add (E item) void set (E item) class Scanner Scanner (InputStream source) Scanner (String str) boolean hasNext() boolean hasNextInt() boolean hasNextDouble() String **next**() int nextInt() double **nextDouble**() String nextLine() Scanner useDelimiter (String regex)

## STANDARD CLASSES AND INTERFACES – SUPPLEMENTAL REFERENCE

Package java.util.function
<pre>Interface BiConsumer<t,u>     void accept(T t, U u)</t,u></pre>
<pre>Interface BiFunction<t,u,r>     R apply(T t, U u)</t,u,r></pre>
<pre>Interface BiPredicate<t,u>   boolean test(T t, U u)</t,u></pre>
<pre>Interface Consumer<t>     void accept(T t)</t></pre>
<pre>Interface Function<t,r>     R apply(T t)</t,r></pre>
<pre>Interface Predicate<t>   boolean test(T t)</t></pre>
Interface Supplier <t> T get()</t>

## **UIL COMPUTER SCIENCE WRITTEN TEST – 2025 REGION**

Note: Correct responses are based on Java SE Development Kit 22 (JDK 22) from Oracle, 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 22 Standard Packages have been imported. Ignore any typographical errors and assume any undefined variables are defined as used. For all output statements, assume that the System class has been statically imported using: import static java.lang.System.\*;

Question 1			
Which of the following is equivalent to the expression 3948 <sub>11</sub> / 222 <sub>4</sub> ?			
<b>A)</b> 243 <sub>7</sub> <b>B)</b> 72 <sub>8</sub>	<b>C)</b> A2 <sub>12</sub>	<b>D)</b> 1330 <sub>4</sub>	E) None are equivalent.
Question 2			
What is output by the code to the right?		out println(212 /	43 - 394 / 63)
(A) -2 $(B) 2$ $(C) 0$ $(C) 0$	<b>))</b> -1	ouc.princin(212 /	10 001 / 00/ <b>/</b>
<b>E)</b> There is no output due to a compile error.			
Question 3			
What is output by the code to the right?		$aut = mintf(\parallel PD \parallel = 17)$	
A) true B) false C) TRUE C	) FALSE	out.princi( %b ,-	⊥/) <b>,</b>
E) There is no output due to a runtime error.			
Question 4		String s = "Infin	itvStone":
What is output by the code to the right?		s.substring(1, s.	length() - 2);
A) nityStoit B) Infinity	Stone207	s += s.charAt(2)	+ s.charAt(5);
C) nitySto221 D) Infinity	Stonefi	<pre>s.substring(3);</pre>	
E) There is no output due to a runtime error.		out.println(s);	
Question 5		boolean a = true ^ true;	
What is output by the code to the right?		a = a & !a    a ^ !a && a   !a;	
A) true B) false		<pre>out.println(a);</pre>	
Question 6			
What is output by the code to the right?		double $g = 5.55;$	
<b>A)</b> 5 <b>B)</b> 6.0 <b>C)</b> 6		int i = Math.roun	d(g);
<b>D)</b> There is no output due to a compile error.		out.println(g);	
E) There is no output due to a runtime error.			
Question 7		int i = 9;	
What is output by the code to the right?		if(i < 10    i++ > 10)	
A) 1211		out.print(1);	. 10)
<b>B)</b> 1311		11(1 < 10 && 1++ )	> IU)
<b>C)</b> 310		else	
<b>D)</b> 1310		<pre>out.print(3);</pre>	
<b>E)</b> 210		out.print(i);	
Question 8		int i = 78 % 13 ^	28;
What is output by the code to the right?		i += 212 - 39   5	1 / 2;
A) 727 B) 81 C) 130 C	) 87	i = 134 & i * 5 +	77;
E) There is no output due to a compile error.		<pre>out.println(i);</pre>	

Question 9	
How many *'s are output by the code to the right?	for(int y=9; y < 123; y=y * 2 - 7)
A) 5 B) 6 C) 7 D) 8	out.print("*");
E) There is no output due to an infinite loop.	
Question 10	int[][] mat = new int[][] {
What is output by the code to the right?	$\{23, 5, -9, 212\},\$
A) [1, 118, 634, 212]	$\{34, 89, 74, 10, 3\},$
<b>B)</b> [33, 124, 74, 10, 3]	{341, 895, 284} \.
<b>C)</b> [0, 118, 634, 212]	for(int i = 0; i < 3; i++) {
<b>D)</b> [34, 122, 74, 10, 3]	<pre>mat[i][i] += mat[i][0];</pre>
E) There is no output due to a runtime error.	<pre>mat[0][i] -= mat[i][i]++; mat[0][i] *= -1;</pre>
	}
	out.println(
Question 11	Arrays.tostring(mat[0]));
What is output by the code to the right?	String k =
A) Koie	"123\nSB Th3\tKoie 8293m";
B) SB Th3	<pre>Scanner sc = new Scanner(k); sc newtline();</pre>
C) Koie 8293m	<pre>sc.next();</pre>
<b>D)</b> Th3	<pre>sc.next();</pre>
<b>E)</b> There is no output due to a runtime error.	<pre>out.println(sc.next());</pre>
Question 12	
What is output by the code to the right?	
<b>A)</b> 245	int sum = 0; for $(int y = 5, y < 16, y = 1)$
<b>B)</b> 185	for (int $x = v$ : $x \ge 10$ ; $y++$ )
<b>C)</b> 266	sum += x;
<b>D)</b> 336	<pre>out.println(sum);</pre>
E) There is no output due to an infinite loop.	
Question 13	
What is the order of precedence for the operators to the right?	I.   (bitwise)
A) II, I, III, IV B) II, III, IV, I	II. ++(pre)
C) III, II, I, IV D) III, II, IV, I	III. *
E) II, III, I, IV	IV.    (IOYICAI)
Question 14	
What is output by the code to the right?	<pre>int[] sizes = new int[] {     Double SIZE Float SIZE</pre>
<b>A)</b> 80	Long.SIZE, Integer.SIZE,
<b>B)</b> 96	Short.SIZE, Byte.SIZE
<b>C)</b> 64	};
<b>D)</b> 128	Arrays.sort(sizes);
E) There is no output due to a runtime error.	<pre>out.print(sizes[3]+sizes[4]);</pre>

Question 15				
What is output by the line mar	ked $//q15$ in the code to the			
right?				
A) [GHIJK, LMNOP]		ArrayList <string> a;</string>		
B) [GHIJK, LMNOP, WXY	Z ]	a.add("ABC");a.add("DEF");		
C) [ABC, DEF, QRS, TU	[ V	<pre>a.add("GHIJK");a.add("LMNOP");</pre>		
D) [ABC, DEF, QRS, TU	V, WXYZ]	a.add("QKS");a.add("10V"); a.add("WXYZ");		
E) There is no output due to a	n infinite loop.	<pre>a.removeIf(s -&gt; s.length() &lt; 4);</pre>		
Question 16		a.forEach(s ->{		
What is output by the line mar right?	ked //q16 in the code to the	<pre>for (int i = 0; i &lt; s.length</pre>	(); i += 2)	
A) HJMOXZ	B) ACDFQSTV	}); //q16		
C) BERU	D) GIKLNPWY			
E) There is no output due to a	compile error.			
Question 17		public void method1(		
Which of the following lines in	the code to the right will be the	int a,	// line 01	
first to create a compile-time e	error?	final int b	// line 02	
<b>A)</b> //line 01	<b>B)</b> //line 02	int $c = a + b;$	// line 03	
<b>C)</b> //line 03	<b>D)</b> //line 04	final int $d = a * b;$	// line 04	
<b>E)</b> //line 05	<b>F)</b> //line 06	out.printf("%d %d %d %d\n",		
<b>G)</b> //line 07	<b>H)</b> //line 08	a, b, c, d);	// line 05	
l) //line 09	<b>J)</b> //line 10	a = a + 1;	// line 06	
K) The code will compile without error but will create a		b = b + 1;	// line 07	
runtime error.		c = c + 1;	// line 08	
L) The code will both compile and run without error.			// IIIe 05	
		out.printf("%d %d %d\n",		
		a, b, c, d); }	// line lu	
Question 18		public void method2(		
Which of the following lines in	the code to the right will be the	ArrayList <integer> a,</integer>	// line 01	
first to create a compile-time e	error?	) {	// 11110 02	
A) //line 01	<b>B)</b> //line 02	ArrayList <integer> c =</integer>	// line 03	
<b>C)</b> //line 03	<b>D)</b> //line 04	final ArrayList <integer> d =</integer>	,,	
<b>E)</b> //line 05	<b>F)</b> //line 06	new ArrayList<>(a);	// line 04	
<b>G)</b> //line 07	<b>H)</b> //line 08	out.printf("%d %d %d %d\n",		
l) //line 09	<b>J)</b> //line 10	a.size(), b.size(),	// line 05	
<b>K)</b> //line 11	<b>L)</b> //line 12		// IIIC 03	
<b>M)</b> //line 13		a.add(1);	// line 06	
N) The code will compile without error but will create a runtime error.		c.add(3); d.add(4);	// line 07 // line 08 // line 09	
<b>O)</b> The code will both compile	and run without error.	a = new ArrayList<>(a);	// line 10	
		c = new ArrayList<>(c);	// line 12	
		<pre>d = new ArrayList&lt;&gt;(d);</pre>	// line 13	
		J		

Question 19	
Which of the following concepts is demonstrated by the $\operatorname{add}$	
methods in the classes to the right?	class A{
A) Overriding	
B) Overloading	private int y;
C) Abstraction	String s;
D) A and B	
E) All of the above.	public A(String str) {
	s = str;
Question 20	y - ,,
Which of the following concepts is demonstrated by the	,
toString method in the classes to the right?	<pre>public int get() {</pre>
A) Overriding	return y;
B) Overloading	}
C) Abstraction	
D) A and B	<pre>public void add() {</pre>
E) All of the above.	y += 3;
	}
Question 21	public String toString() {
What is output by the line marked $//q21$ in the code to the	return s;
right?	}
<b>A)</b> 7	
<b>B)</b> 10	}
<b>C)</b> 13	
<b>D)</b> There is no output due to a compile error.	CLASS B EXTENDS A{
<b>E)</b> There is no output due to a runtime error.	int v•
Question 22	<pre>public B(String str, int i) {</pre>
What is output by the line marked $//q22$ in the code to the	<pre>super(str);</pre>
right?	y = i;
<b>A)</b> 7	}
<b>B)</b> 12	nublic word add(int i) (
<b>C)</b> 24	v += i
<b>D)</b> There is no output due to a compile error.	}
E) There is no output due to a runtime error.	
	}
Question 23	/////////client code//////////
What is output by the line marked $//{\rm q23}$ in the client code to	A = new A("A");
the right?	B b = new B("B", 12);
<b>A)</b> AB	a.auu(); out println(a get()), //g21
<b>B)</b> BA	b.add(12);
<b>C)</b> Output cannot be determined until runtime.	out.println(b.get()); //q22
<b>D)</b> There is no output due to a compile error.	out.println(a+b); //q23
E) There is no output due to a runtime error.	

Question 24		
Which of the following could replace <1*> in the code to the	Class Jedi {	
right and have the code compile without error (including the	<pre>public static void speak() {</pre>	
client code)?	out.println("May the force	
A) public B) final	be with you.");	
C) protected D) static	}	
E) Replace all instances of <1*> with nothing (delete it and leave the space blank).	<1*> class ObiWan {	
Question 25	public static word speak() (	
Assuming that <1*> has been filled in properly in the code to	out println("Hello	
the right, what is output by the client code to the right?	there!");	
A) Hello there!	}	
May the force be with you.		
B) May the force be with you. Hello there!	}	
C) Hello there!	}	
Hello there!		
D) May the force be with you.	//////// Client Code /////////	
May the force be with you.	Jedi.Oblwan.speak();	
E) The output cannot be determined until runtime.	Jear.spear(),	
Question 26		
Which of the following is equivalent to the logic circuit diagram to the right?		
A) $\overline{\overline{A} + \overline{B}} \oplus \overline{C * D}$		
<b>B)</b> $\bar{A} + \bar{B} \bigoplus \overline{C + D}$		
$\mathbf{C} \ \overline{\overline{A} * \overline{B}} \oplus \overline{C + D}$		
D) More than one of the above.		
E) None of the above.		
Question 27		
How many permutations of true and false assignments to		
A, B, C, and D output true in the circuit diagram to the right?		
<b>A)</b> 2 <b>B)</b> 4 <b>C)</b> 7 <b>D)</b> 10 <b>E)</b> 13		
Question 28		
Suppose we replaced input <i>D</i> with a second instance/pin/line of input <i>C</i> . How many permutations of true and false assignments to <i>A</i> , <i>B</i> , and <i>C</i> output true in the modified circuit diagram proposed?	$D \longrightarrow$	
A) 2 B) 4 C) 5 D) 7 E) 8		

```
Question 29
                                                        abstract class Agent {
What is the correct compile-time error with the code to the
                                                             public final String name;
right as well as the appropriate way to resolve the error?
                                                             public Agent(String name) {
A) They keyword final is not a valid modifier to a class
  definition. To resolve, remove all final modifiers to the
                                                                   this.name = name;
  class definition.
                                                              }
B) Abstract classes can only define abstract methods and
  cannot define abstract fields. To resolve, remove the
                                                             public abstract void ult();
  declaration of the name variable and the constructor from
                                                        }
  the abstract class and add the same variable definition to
  the three sub-classes. Then, instead of calling the parent
                                                        final class Tejo extends Agent {
  constructor using super, instead assign the (now) local
                                                             public Tejo() {
  field name to the value previously passed using super.
                                                                    super("Tejo");
C) Class definitions modified with the keyword final
  prevents other classes from inheriting from it. To resolve,
  remove the class definition of EnemyJett, as the client
                                                             public void ult() {
  code does not rely on this class existing. Alternatively, if we
                                                                    out.printf("%s: This is
  needed the EnemyJett class to exist, have the
                                                                         how it ends!\n",
  EnemyJett class instead inherit from the Agent abstract
                                                                         name);
  class instead of the Jett class.
                                                              }
D) Classes that are declared as abstract, if inherited, need
                                                        }
  to use the implements keyword instead of the extends
  keyword. To resolve, replace all instances of extends with
                                                        final class Jett extends Agent {
  implements.
                                                             public Jett() {
E) Abstract classes can never appear as the datatype of a
                                                                    super("Jett");
  variable declaration – only concrete classes or primitive
                                                              }
  datatypes can. To resolve, replace the datatype of the tejo
  variable with Tejo, and replace the datatype of the jett
                                                             public void ult() {
  variable with the Jett object.
                                                                    out.printf("%s: Watch
Question 30
                                                                         this!\n", name);
Assuming that the correct compile-time error identified above
                                                              }
has been resolved using its correct resolution method, what is
                                                        }
output by the client code to the right?
A) Tejo: This is how it ends!
                                                        class EnemyJett extends Jett {
  Jett: Watch this!
                                                             public EnemyJett() {
B) Tejo: This is how it ends!
                                                                    super("Jett");
  Jett: Get out of my way!
                                                              }
C) Tejo: This is how it ends!
  Jett: Watch this!
                                                             public void ult() {
  Jett: Get out of my way!
                                                                    out.printf("%s: Get out of
D) There is no output, despite the program compiling and
                                                                         my way!\n", name);
  running without error.
                                                              }
E) The output cannot be determined until runtime.
                                                        }
                                                        /////// Client Code /////////
                                                        Agent tejo = new Tejo();
                                                        Agent jett = new Jett();
                                                        tejo.ult();
                                                        jett.ult();
```

	stion 31	Ċ
Wh	at can replace <1*> in the code to the right so that any	
obje	ect which implements the Comparable interface can be	
stor	ed within the new object DataStructure?	
A)		
B) (	Comparable	
C)	? extends Comparable	
D) '	I extends Comparable <t></t>	
E)	None of the above	
Que	stion 32	
Ass rep imp the	uming that <1*> has been filled in properly, what can lace <2*> in the code to the right so that any object which lements the Comparable interface can be stored within new object DataStructure?	
A)	Γ	
B)	Comparable	
C)	? extends Comparable	
D)	I extends Comparable <t></t>	
E)	None of the above	
Que	stion 33	
wn ext	at is the tightest upper bound on the time complexity of the	
Δ)	$\mathcal{O}(n)$	
R)	$\mathcal{O}(n \log_2 n)$	
	$\sigma(\sqrt{n})$	
	O(1)	
D) (	$\mathcal{O}(1)$	
E) (	$\mathcal{O}(\log_2 n)$	
Que	stion 34	
The	class DataStructure in the code to the right is an	
imp	lementation of what well-known data structure?	
A)	Dequeue	
B) /	AVL Tree	
C)	Binary Search Tree	
D)	Min-Heap	
	Max-Heap	
<b>E)</b>		
E)		

```
ss DataStructure<<1*>> {
private ArrayList<<2*>> arr;
public DataStructure(int n) {
   arr = new ArrayList<<2*>>();
 }
private void swap(int i, int j) {
   <2*> temp = arr.get(i);
   arr.set(i, arr.get(j));
   arr.set(j, temp);
 }
public void insert(<2*> val) {
   int i = arr.size();
    int parent = (i - 1) / 2;
    arr.add(val);
    while (i > 0 && arr.get(i).compareTo(
       arr.get(parent)) < 0)</pre>
    {
       swap(i, parent);
       i = parent;
       parent = (i - 1) / 2;
    }
 }
public <2*> extract() {
   if (arr.isEmpty()) {
       return null;
    }
    <2*> min = arr.get(0);
    <2*> last = arr.remove(arr.size() - 1);
    if (!arr.isEmpty()) {
       arr.set(0, last);
       int i = 0;
       while (true) {
          int l = (2 * i) + 1;
          int r = (2 * i) + 2;
          int minIndex = i;
          if (l < arr.size() && arr.get(l)</pre>
             .compareTo(arr.get(minIndex))
             < 0)
          {
             minIndex = 1;
          if (r < arr.size() && arr.get(r)</pre>
             .compareTo(arr.get(minIndex))
             < 0)
          {
             minIndex = r;
          }
          if (minIndex == i) {
             break;
          }
          swap(i, minIndex);
          i = minIndex;
       }
    }
    return min;
 }
```

#### Question 35

Given a sorted array of  $m = n^2$  elements, in terms of n, which of the following is the tightest upper bound for the fastest way to find the index of a specific element <u>among the *m* elements</u>?

A) 
$$\mathcal{O}(m) = \overline{\mathcal{O}(n^2)}$$
  
B)  $\mathcal{O}(m) = \mathcal{O}(\sqrt{n^2}) = \overline{\mathcal{O}(n)}$   
C)  $\mathcal{O}(m) = \mathcal{O}((\log n^2)^2) = \mathcal{O}((2\log n)^2) = \mathcal{O}(4\log^2 n) = \overline{\mathcal{O}(\log^2 n)}$   
D)  $\mathcal{O}(m) = \mathcal{O}(\log(n^2)) = \mathcal{O}(2\log n) = \overline{\mathcal{O}(\log n)}$   
E)  $\mathcal{O}(m) = \overline{\mathcal{O}(1)}$ 

#### Question 36

Given a sorted singly-linked list of  $m = n^2$  elements, in terms of n, which of the following is the tightest upper bound for the fastest way to find the index of a specific element <u>among the *m* elements</u>?

A) 
$$O(m) = \overline{O(n^2)}$$
  
B)  $O(m) = O(\sqrt{n^2}) = \overline{O(n)}$   
C)  $O(m) = O((\log n^2)^2) = O((2 \log n)^2) = O(4 \log^2 n) = \overline{O(\log^2 n)}$   
D)  $O(m) = O(\log(n^2)) = O(2 \log n) = \overline{O(\log n)}$   
E)  $O(m) = \overline{O(1)}$   
Cuestion 37  
The code to the right demonstrates which of the following design patterns?  
A) Singleton  
B) Object Pool  
C) Builder  
D) Prototype  
E) Factory Method.  
C) Builder  
B) Prototype  
E) Factory Method.  
C) Builder  
B) Merge Sort  
C) Quick Sort  
B) Merge Sort  
C) Quick Sort  
C) Quick Sort  
C) Quick Sort  
C) Quick Sort  
D) Option B and C.  
E) None of the above.  
C) Quick Sort  
C) Quick Sort

single-source shortest path problem. In terms of V, the number of vertices in the graph, and E, the number of edges in the gra what is the tightest asymptotic upper bound for solving this problem? Express your answer in Big-O notation.

Question 40	The stack is initially empty.
What is the sum of all elements that remain in the stack after	Push 18
the process denoted to the right completes?	Push 12
	Push 25
	Push 19
	Рор
	Рор
	Push 17
	Push 32
	Push 36
	Push 24
	Peek
	Peek
	Pop
	Push 56
	Push 17
	Pop
	Process Complete

## $\star$ ANSWER KEY – CONFIDENTIAL $\star$

## UIL COMPUTER SCIENCE – 2024-2025 REGION

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

1) <u> </u>	11) <u>A</u>	21) <u> </u>	31) <u>D</u>
2) <u> </u>	12) <u> </u>	22) <u> </u>	32) <u> </u>
3) <u> </u>	13) <u> </u>	23) <u>D</u>	33) <u>E</u>
4) <u> </u>	14) <u>B</u>	24) <u>D</u>	34) <u>D</u>
5) <u> </u>	15) <u> </u>	25) <u>A</u>	35) <u>D</u>
6) <u>D</u>	16) <u>D</u>	26) <u>C</u>	36) <u>A</u>
7) <u>D</u>	17) <u> </u>	27) <u>D</u>	37) <u> </u>
8) <u> </u>	18) <u> </u>	28) <u> </u>	38) <u> </u>
9) <u> </u>	19) <u> </u>	29) <u> </u>	*39) <u>O(V²) or O(E)</u>
10) <u> </u>	20) <u>A</u>	30) <u>A</u>	*40) 171

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

Note: Correct responses are based on Java SE Development Kit 22 (JDK 22) 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 22 Standard Packages have been imported. Ignore any typographical errors and assume any undefined variables are defined as used.

#### Explanations:

1.	С	Simple base conversion (Convert to base 10 then simple division, convert answers as well).
2.	А	Simple expression solving (PEMDAS)
3.	С	%B will output FALSE if the parameter is null or the literal false, otherwise it will output TRUE.
4.	В	Strings are immutable, so both substring calls do not affect the original string, and the
		characters will be converted to $ints$ and added together before they are appended to the end
		of the string.
5.	A	Simple boolean expression.
6.	D	Math.round(double) will return a long, so there is a compile error trying to store that value in an int.
7.	D	Simple if else solution, but    and && short circuit if the first expression guarantees the
		entire expression is false, and i++ will return the original value then increment it afterwards.
8.	С	Simple expression solving, order of operations will play a part
9.	В	Simple output loop tracing.
10.	C	Simple matrix and loop tracing.
11.	А	Scanner tracing, nextLine will get all data up to a \n, next will grab all characters up to the
		next whitespace character.
12.	А	Simple loop tracing.
13.	E	Java order of precedence.
14.	В	Need to know that arrays are 0-indexed, and all of the sizes of the data types listed.
15.	В	The removelf does exactly what it sounds like, and the list will contain all of the elements with
		length greater than or equal to 4.
16.	D	The forEach method will print out every other item in the list, starting at position 0.
17.	G	When a variable storing a primitive datatype is assigned the keyword final, its contents can be
		read; however, its value cannot be changed after its initial declaration. Any parameter of a
		method can be declared as final and, if the datatype is also primitive, then the same rule
		applies. The first time that the code attempts to change the value of a final primitive is the line
		marked //line 07 and will create a compile-time error.
18.	К	When a variable storing an object is assigned the keyword final, the only action that cannot
		be performed on that variable is re-assigning what object the variable points to. Thus,
		attempting to modify the underlying structure is actually allowed. However, the first time that
		the code attempts to change what object a final variable is pointing to is the line marked
10		//line ll and will create a compile-time error.
19.	В	Overloading is correct, because the add () method is inherited from class A into class B, so
		there are two add methods with different parameter combinations. See this definition of
20	^	The test the method of the Object close is everyidden by close 7
20.	A	The ust method will return the value 10 from a (the instance of class 3) which began with 7
21.	В	and added 2
22	^	The $x \to t$ method will get 7 from the underlying instance of class $\lambda$ that b (the instance of class
22.	~	B) is built on This 7 is not affected by the call to add, since add will only affect the instance in
		class B. The reason the 7 is returned is because it is a $private$ instance variable in the class
		where $aet$ () was defined and lava gives this higher precedence than the same named variable
		in the subclass, even when the method is inherited.
23.	D	You cannot add these types together, if you want both toString methods to be accessed.
	_	you'd need to have " "+a+b instead.
24.	D	Applying the keyword static to a nested class makes it associated with its outer class, but
		does not require an instance of the outer class to be able to access. Since the client code
		attempts to access the ObiWan class without creating an instance of the Jedi class, the
		keyword static is required.

25.	А	Note that the two speak methods are not linked in any way (as you might have with
		inheritance) and therefore, the first line calls the ObiWan class's speak method, and the
		second line calls the Jedi class's speak method.
26.	C	Both A and B are fed into a NOT gate, which are both fed into a NAND gate. C and D are fed into
		a NOR gate. The output of the NAND and the NOR gate are fed into an XOR gate.
27.	D	The following is a copy of the truth table for the expression in the logic diagram:
		$A \mid B \mid C \mid D \mid \overline{\overline{A} * \overline{B}} \oplus \overline{C + D}$
		F F F F T
		Of the 16 permutations, 10 are true and 8 are false.
28.	В	The following is a copy of the truth table for the expression proposed in question 32:
		$A \mid B \mid C \mid \overline{\overline{A} * \overline{B}} \oplus \overline{\overline{C} + C}$
		F F T F
		Of the 8 permutations, 4 are true, and 4 are false.
29.	С	This option is the only one that correctly acknowledges both the error with the code, as well as a
		valid/acceptable method of resolving the error.
30.	А	Assuming that we used one of the two resolution methods mentioned in option C from question
		28, then the client code will first call the ${\tt Tejo}$ class's <code>ult</code> method, and then the <code>Jett</code> class's
		ult method. There is no error in referencing the name field of the Agent class without the
		super modifier (otherwise, an option about this would have been written for question 28 since
		there is only one error as mentioned).
31.	D	This will require the use of generics, hence the T. Moreover, since the problem asks that objects
		implement the Comparable interface, we need to append extends Comparable <t>.</t>
32.	A	Note that we don't have to specify $T$ extends Comparable $T$ for every other reference
		after having declared the properties of $T$ . Afterwards, we can just refer to the generic type $T$ ,
		since the compiler knows that I implements the Comparable interface by what we did in
33	F	f yuesului pp.
55.		that it runs in $O(\log_n)$ time. Note that the loop will run at most $O(\log_n)$ time since after
		each iteration, we cut down the search space by a factor of 2
34.	D	As previously stated, this is a heap, but since the element removed by the heap when the extract
-	_	method is called is the smallest value, that makes it a min-heap. Note that all data structures
		mentioned (except for option A) do have a tree-like structure, the only one that satisfies all
		properties of the data structure is a min-heap.

35.	D	With an array, we can do $O(1)$ random access of the list, which means that we can perform binary search on the list. Binary search takes $O(\log_2 n)$ time, where <i>n</i> is the array size. The input size is $m = n^2$ , so the total complexity is $O(\log_2 n^2) = O(2\log_2 n) = O(\log_2 n)$
36.	A	With a singly-linked list, we cannot do $O(1)$ random access, and have to do a linear search for elements. Since the input size is $m = n^2$ , the total time complexity is $O(n^2)$ .
37.	A	Singleton is a creational design pattern that lets you ensure that a class has only one instance, while providing a global access point to this instance.
38.	С	Despite having an average time complexity of $\mathcal{O}(n \log_2 n)$ , Quick Sort is the only algorithm among the options that has a worst-case time complexity of $\mathcal{O}(n^2) \neq \mathcal{O}(n \log_2 n)$ .
39.	$\mathcal{O}(V^2)$ or $\mathcal{O}(E)$	Note that since the graph is complete/dense, the best backing structure to use here isn't a heap (which would work in $\mathcal{O}((V + E) \log_2 V)$ time for a Binary Heap or $\mathcal{O}(V \log_2 V + E)$ time for a Fibonacci Heap) but instead a 2D array / matrix / adjacency list. Note that for a complete graph, $ E  \equiv  V ^2$ , so either $\mathcal{O}(V^2)$ or $\mathcal{O}(E)$ are acceptable answers, despite meaning two different things in general.
40.	171	This is equivalent to the following sum: $18 + 12 + 17 + 32 + 36 + 56 = 171$