UIL COMPUTER SCIENCE WRITTEN TEST

2022 DISTRICT

MARCH 2022

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 isEmpty()
  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

Interface BiConsumer<T,U>
```
  void accept(T t, U u)
```

Interface BiFunction<T,U,R>
```
  R apply(T t, U u)
```

Interface BiPredicate<T,U>
```
  boolean test(T t, U u)
```

Interface Consumer<T>
```
  void accept(T t)
```

Interface Function<T,R>
```
  R apply(T t)
```

Interface Predicate<T>
```
  boolean test(T t)
```

Interface Supplier<T>
```
  T get()
```

UIL COMPUTER SCIENCE WRITTEN TEST – 2022 DISTRICT

Note: Correct responses are based on **Java SE Development Kit 17 (JDK 17)** 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 17 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 equal to DC_{16} - BD_{16}?

A) 21_{16} **B)** 47_{10} **C)** 11110_2 **D)** $1F_{16}$ **E)** None of these.

Question 2.

What is the output of the code segment to the right?

A) `1`

B) `49`

C) `28`

D) `28.0`

E) There is no output due to a syntax error

```
out.print(14 + -3 - +3 * 7 / 2);
```

Question 3.

What is the output of the code segment to the right?

A)
```
PI
3.14Pie
```

B)
```
Pi
3.141Pie
```

C)
```
Pi
3.142Pie
```

D)
```
Pi
3.141
Pie
```

E)
```
Pi
3.142
Pie
```

```
out.println("Pi");
out.printf("%.3f",3.14159);
out.println("Pie");
```

Question 4.

What is the output of the code segment to the right?

A) `1`

B) `2`

C) `3`

D) `5`

E) `-1`

```
out.print("banana".lastIndexOf("a",3));
```

Question 5.

What is the output of the code segment to the right?

A) `true`

B) `false`

```
boolean t = true;
boolean f = false;
boolean tt = true;
boolean ff = false;
out.print(t ^ f ^ tt ^ ff);
```

Question 6.

What is the output of the code segment to the right?

A) `-1.0`

B) `0.0`

C) `1.0`

D) `1`

E) `3.0`

```
out.print(Math.signum(Math.PI));
```

Question 7.

What is the output of the code segment to the right?

A) `-337.5`

B) `-497.5`

C) `52.5`

D) `84.5`

E) There is no output due to an error

```
int x = -5;
double y = 2.5;
char z = 'a';
out.print(x * y + z);
```

Question 8.

What is the output of the code segment to the right?

A) `small x`

B) `big X`

C) `small X`

D) `crazy`

E) There is no output due to a compile error

```
char ch = 'X';
char space = ' ';
if(ch % 2 == 0)
{
  ch += space;
}
if(Character.isLowerCase(ch))
{
 out.print("small "+ ch);
}else if(Character.isUpperCase(ch)){
 out.print("big "+ ch);
}else{
 out.print("crazy");
}
```

Question 9.

How many asterisks are printed by the code shown to the right?

A) 13

B) 24

C) 26

D) 28

E) 52

```
for(int x = 17; x <= 41; x += 2)
{
  out.print("**");
}
```

Question 10.

What is the output of the code segment to the right?

A) `[3, 1, 4, 1, 5, 9]`

B) `[3, 1, 4, 4, 1, 3]`

C) `[9, 5, 1, 1, 5, 9]`

D) `[5, 1, 6, 4, 0, 5]`

E) There is no output due to an error.

```
int [] ints = {9, 5, 1, 4, 1, 3};
ints[0] = ints[ints.length - 1];
ints[ints.length - 1] = ints[0];
ints[1] = ints[ints.length - 2];
ints[ints.length - 2] = ints[1];
ints[2] = ints[ints.length - 3];
ints[ints.length - 3] = ints[2];
out.print(Arrays.toString(ints));
```

```
public static void main(String[] args) throws IOException
{
      Scanner f = new Scanner(new File("data.dat"));
      <code>;
      String s = "";
      int t = 0;
      while(f.hasNext())
            {
                  s += f.next();
                  t++;
            }
      out.print(t);
      f.close();
}
```

Question 11.

Consider the main method shown above. Suppose `data.dat` contains: "**`car,truck,train,plane`**". What could replace **<code>** so the code outputs `4`?

A) `f.setDelimiter(",");`

B) `f.useDelimiter(",");`

C) `f.separateByCommas();`

D) `f.delimiter(",");`

E) None of the above

Question 12.

What is the output of the code segment to the right?

A) `127`

B) `1364`

C) `341`

D) `340`

E) `None of the above`

```
int y = 0;
for(int x = 1;x<=16;x*=2)
{
  y += x*x;
}
out.print(y);
```

Question 13.

What is the correct order of operations for the operators listed on the right?

A) I II III

B) II III I

C) III II I

D) II I III

E) III I II

I. >>

II. &

III. +

Question 14.

What is the output of the code segment to the right?

A) -1.0

B) 0.0

C) 1.0

D) There is no output due to a compile error

E) There is no output due to a runtime error

```
out.print(Math.signum(Double.MIN_VALUE));
```

Question 15.

What is the output of the code segment to the right?

A) 3

B) 6

C) 8

D) 9

E) 11

```
ArrayList<Integer> list;
list = new ArrayList<Integer>();
for(int i = 0;i<18;i++)
{
   list.add(i * 5 % 7);
}
list.remove(4);
list.remove(12);
out.print(list.get(8) + list.get(14));
```

Question 16.

What is the output of the code segment to the right?

A) 4 **B)** 5 **C)** 7 **D)** 8

E) There is no output due to a compile error

```
int x = 7;
int y = 4;
if((x & 9) == 1 & y++ == 4)
{
   out.println(y);
}else{
   out.println(x);
}
```

Question 17.

Which of the following data structures' add operation has the highest worst case Big O complexity?

A) LinkedList

B) TreeSet

C) PriorityQueue

D) Stack

E) ArrayList

Question 18.

Which of the following is the *best estimate* of the output of the code segment shown on the right?

A) 0.25

B) 0.5

C) 0.75

D) 1

E) 5000

```
Random r = new Random();
double ct = 0;
double it = 10000;
for(int i=1;i<=it;i++)
{
   double x = r.nextDouble();
   double y = r.nextDouble();
   if(x*x < y*y)
      ct++;
}
out.print(ct / it);
```

Question 19.

What is the output of the code segment to the right?

A) 0

B) 2

C) 3

D) 4

E) 8

```
int x = Integer.parseInt("1024", 5);
out.print(Integer.bitCount(x));
```

Question 20.

What is the output of the code segment to the right?

A) true

B) false

C) [1, 2, 3, 4, 5]

D) There is no output due to a compile error

E) There is no output due to a runtime error

```
HashSet<Integer> s1 = new HashSet<>();
HashSet<Integer> s2 = new HashSet<>();
int[] ints = new int[]{1,2,3,4,5};
for(int i = 0;i<5;i++)
{
   s1.add(ints[i]);
   s2.add(ints[ints.length - 1 - i]);
}
out.print(s1.equals(s2));
```

Question 21.

What is the output of the code segment to the right?

A) [0, 2, 5, 7, 8, 11]

B) [11, 8, 7, 5, 2, 0]

C) [5, 8, 2, 7, 0, 11]

D) [11, 0, 7, 2, 8, 5]

E) There is no output due to a compile error

```
Stack<Integer> st = new Stack<>();
int[] ints = new int[]{5,8,2,7,0,11};
for(int x: ints)
   st.add(x);
Collections.sort(st);
out.print(st);
```

Question 22.

What is returned by the function call below?

```
rec("UIL", "WIN", 0, 0);
```

A) 0

B) 1

C) 2

D) 3

E) 4

Question 23.

What is returned by the function call below?

```
Dp("Door", "Ford");
```

A) 0

B) 1

C) 2

D) 3

E) 4

Question 24.

Assume that a and b are strings of length N. Which of the following is the most restrictive Big O upper bound of method call `rec(a, b, 0, 0)`?

A) $O(N)$

B) $O(N^2)$

C) $O(N^3)$

D) $O(2^N)$

E) None of the above

Question 25.

Assume that a and b are strings of length N. Which of the following is the most restrictive Big O upper bound of method call `dp(a, b)`?

A) $O(N)$

B) $O(N^2)$

C) $O(N^3)$

D) $O(2^N)$

E) None of the above

```
public static int rec(String a, String b,
                                int ai, int bi){
  if(ai == a.length() ||
     bi == b.length()){
     return 0;
  }
  if(a.charAt(ai) == b.charAt(bi))
  {
     return 1 + rec(a,b,ai+1,bi+1);
  }
  int x = rec(a,b,ai+1,bi);
  int y = rec(a,b,ai,bi+1);
  return Math.max(x,y);
}

public static int dp(String a, String b)
{
   int al = a.length();
   int bl = b.length();
   int[][] dp = new int[al+ 1][bl + 1];
   for(int i = 0;i<al;i++)
   {
      for(int j = 0;j<bl;j++)
      {
         if(a.charAt(i) == b.charAt(j))
         {
            dp[i+1][j+1] = dp[i][j] + 1;
         }else{
            int x = dp[i][j+1];
            int y = dp[i+1][j];
            dp[i+1][j+1] = Math.max(x,y);
         }
      }
   }
   return dp[al][bl];
}
```

Question 26.

What is the output of the code segment to the right?

A) `100`

B) `0100`

C) `1210`

D) `1100100`

E) None of the above

```
out.println(Integer.toString(100,4));
```

Question 27.

How many constructors does the `Cloud` class have?

A) `0`

B) `1`

C) `2`

D) `3`

E) `4`

Question 28.

What is the output of client code shown below?

```
Cloud a = new Cloud(35, 35, 1000);
Cloud b = new Cloud(20, 51, 58);
out.println(a);
```

A) `1225 1000 cirrus`

B) `1225 1000 stratus`

C) `1225 1000 cumulus`

D) `1225 1000 cumulonimbus`

E) There is no output due to an error.

```
public class Cloud{

  int volume;
  int density;
  int alt;

  public Cloud(int v, int d, int a)
  {
    this.volume = v;
    this.density = d;
    this.alt = a;
  }

  public int weight() {
    return volume * density;
  }

  public String type(){
    if(weight() > 1000 && alt < 100)
    {
      return "cumulonimbus";
    }
    if(alt > 500)
    {
      return "cirrus";
    }else {
      return "cumulus";
```

Question 29.

What is the output of client code shown below?

```
Cloud a = new Cloud(35, 35, 1000);
Cloud b = new Cloud(20, 51, 58);
out.println(b.dangerous());
```

A) 1020 58 cumulonimbus

B) 1020 58 cumulus

C) true

D) false

E) There is no output due to an error.

```
    }
  }

  public boolean dangerous()
  {
    return type().equals("cumulonimbus");
  }

  public String toString()
  {
     return weight()+" "+alt+" "+type();
  }

}
```

Question 30.

How many swaps would insertion sort perform on the array [5, 4, 2, 3, 1] to sort into ascending order?

A) 1

B) 3

C) 5

D) 9

E) 12

Question 31.

What is the output of the client code below?

```
FI lambda = i -> i % 10 + i / 10;
out.print(lambda.go(12345));
```

A) 12345

B) 12350

C) 1239

D) There is no output due to a compile error

E) There is no output due to a runtime error

```
public class Lambdas{
  private interface FI
  {
    public int go(int i);
  }
}
```

Question 32.

If it takes Bubble Sort 5 seconds to sort 4 million elements, what is the best estimate of how long will it take to sort 16 million elements?

A) 20 seconds

B) 25 seconds

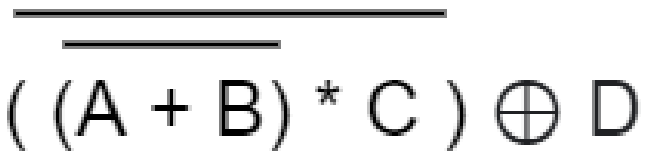
C) 64 seconds

D) 80 seconds

E) 100 seconds

Question 33.

How many assignments of variables A,B,C, and D make the expression to the right evaluate to true?



A) 0

B) 1

C) 7

D) 8

E) 15

Question 34.

The letters in "TEXASUIL" are inserted into a binary search tree in order. What is the post-order traversal of the resultant tree?

A) AEILSTUX

B) TEXASUIL

C) ALISEUXT

D) AEILSTUX

E) None of the above

Question 35.

Which of the following is used to denote the beginning of a line in regular expressions?

A) %

B) $

C) ^

D) \b

E) None of the above

Question 36.

What is the output of the client code below?

```
Structure<Integer> s;
s = new Structure<Integer>(6);
int N = 314159265;
while(N != 0)
{
  s.add(N % 10);
  N /= 10;
}
out.println(s.size());
```

A) 6

B) 7

C) 9

D) The output will vary from run to run

E) There is not output due to an error

```
public class Structure<T>{
   int cap;
   int sz;
   HashMap<Integer, LinkedList<T>> map;

   public Structure(int cap)
   {
     this.cap = cap;
     map = new HashMap<>();
   }

   public void add(T t)
   {
     if(contains(t))
       return;
     int i = t.hashCode() % cap;
     if(!map.containsKey(i))
       map.put(i, new LinkedList<T>());
     map.get(i).add(t);
     sz++;
```

Question 37.

Which of the following is the most restrictive Big O upper bound of method `add`?

A) O(1)

B) O(log(N))

C) O(N)

D) $O(N^2)$

E) None of the above

Question 38.

What data structure does the `Structure` class represent?

A) Map

B) HashMap

C) Queue

D) ArrayList

E) HashSet

```
    }

    public boolean contains(T t)
    {
      int i = t.hashCode() % cap;
      if(!map.containsKey(i))
      return false;
      for(T x: map.get(i))
        if(x.equals(t))
          return true;
      return false;
    }

    public int size()
    {
      return sz;
    }
}
```

Question 39.

What could replace **<code>** for the code to the right to output `progaming`?

Write your answer in the blank provided on the answer document.

```
int x = <code>;
if(x >= 15)
{
  out.print("pro");
}
if(x <= 20)
{
  out.print("g");
}
if(x % 2 == 0)
{
  out.print("r");
}
if(x % 5 != 0)
{
  out.print("am");
}
if(x % 10 == 9)
{
  out.print("m");
}
out.print("ing");
```

Question 40.

If a binary tree contains 12 nodes, what is the most number of levels (including the root) that tree might have? Write your answer in the blank provided on the answer document.

★ANSWER KEY – CONFIDENTIAL★

UIL COMPUTER SCIENCE – 2022 DISTRICT

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

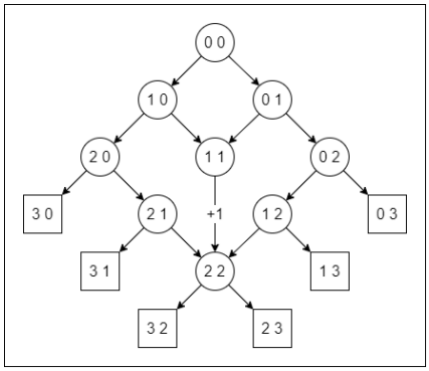
1) D	11) B	21) A	31) C
2) A	12) C	22) B	32) D
3) C	13) E	23) C	33) D
4) C	14) C	24) D	34) C
5) B	15) B	25) B	35) C
6) C	16) B	26) C	36) B
7) D	17) E	27) B	37) C
8) A	18) B	28) A	38) E
9) C	19) D	29) C	*39) 17
10) B	20) A	30) D	*40) 12

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

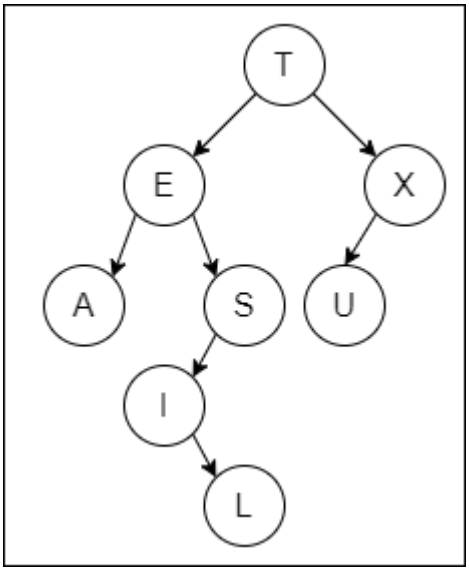
Note: Correct responses are based on **Java SE Development Kit 17 (JDK 17)** from Oracle, Inc. All provided code segments are intended to be syntactically correct, unless otherwise stated (e.g., "error" is an answer choice) and any necessary Java SE 17 Standard Packages have been imported. Ignore any typographical errors and assume any undefined variables are defined as used.

Explanations:

1.	D	DC_{16} - BD_{16} = $1F_{16}$
2.	A	14 – 3 – (3 * 7 / 2) = 14 – 3 – 10 = 1
3.	C	%.3f rounds to 3 places after the decimal, and printf does not print a new line
4.	C	lastIndexOf starts searching backwards at position 3, immediately finding an a
5.	B	An even number of trues and falses in an xor chain will always evaluate to false
6.	C	Signum returns a double representing the sign of a given number. Pi is positive, so signum returns 1.0
7.	D	(-5.0 * 2.5) + 'a' = -12.5 + 97 = 84.5
8.	A	'X' = 120, 'X' – ' ' = 'x', so 'small x' is output
9.	C	The loop executes 13 times, adding ** to the output on each iteration. Hence 26 asterisks
10.	B	[9, 5, 1, 4, 1, 3] [3, 5, 1, 4, 1, 3] [3, 5, 1, 4, 1, 3] [3, 1, 1, 4, 1, 3] [3, 1, 1, 4, 1, 3] [3, 1, 4, 4, 1, 3] [3, 1, 4, 4, 1, 3]
11.	B	The default delimiter for the next() method is whitespace. There is not any white space in the input string, so by default the code would output 1. By using SetDelimiter to delimit by commas instead, an output of 4 can be achieved.
12.	C	1 * 1 + 2 * 2 + 4 * 4 + 8 * 8 + 16 * 16 = 1 + 4 + 16 + 64 + 256 = 341
13.	E	Addition comes before all bitwise operations. Bitwise operations happen in SHAXOR order (shift, and , xor, or)
14.	C	Double.MIN_VALUE is the closest positive double to zero. Because it is positive, signum will return 1.0
15.	B	Due to the removal of 4 and 12, the original positions of 8 and 14 are 9 and 16, respectively. 9 * 5 % 7 + 16 * 5 % 7 = 3 + 3 = 6
16.	B	0111 & 1001 = 1, and the y++ increment takes place after the comparison, so the if statement evaluates to true. Y is then 5 due to the increment
17.	E	ArrayList add has an O(n) worst case big O complexity for add. Every other structure option was either O(1) worst case or O(log(n))
18.	B	x*x < y*y is equivalent to x < y. if x and y are picked at random, x will be less than y around half of the time, so .5 is the best estimate.
19.	D	1024_5 = 139 = 10001011_2 for a bitcount of 4
20.	A	set .equals method compares content regardless of any ordering. all 5 numbers are present in each set, so .equals will return true
21.	A	stacks can indeed be sorted, and they print their contents first to last

22.	B	22 - 25 deal with an implementation of longest common subsequence. Below is each possible path the recursion could follow. Squares represent base cases where 0 is returned. Edges with +1 represent cases where the second if is triggered. The maximum of any path is going to go a single +1 edge, as only one exists. This is analogous to saying the two words only share one letter, so Longest Common Subsequence cannot exceed 1. 

23.	C	The dynamic programming table will look like this: 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 1 1 2 0 0 1 1 2 Representing a LCS of “or”, of length 2
24.	D	Each recursive call to rec spawns 2 new calls, and this occurs to a recursive depth of N. Hence runtime is 2^N
25.	B	Each value in the table is filled by referencing a constant number of other table values. the table is N*N, so runtime is N^2
26.	C	$1210_4 = 100$
27.	B	Cloud has exactly one constructor
28.	A	altitude is too high to be cumulonimbus, so type() returns cirrus
29.	C	B is cumulonimbus, so dangerous() returns true.
30.	D	[4, 5, 2, 3, 1] 1 swap [2, 4, 5, 3, 1] 2 swaps [2, 3, 4, 5, 1] 2 swaps [1, 2, 3, 4, 5] 4 swaps total of 9 swaps
31.	C	This is a syntactically correct functional interface and lambda function declaration. The function takes an input i and returns i % 10 + i / 10. For 12345, this evaluates to 1239
32.	D	Bubble sort is an $O(N^2)$ sort, so multiplying size by 4 would expect to multiply runtime by 4^2. 5 * 4 * 4 = 80 seconds.

33.	D	For any assignment between A, B, and C, D being 1 will produce one result, and D being 0 will produce the other. The number of true assignments must equal the number of false assignments, so the answer is 8.
34.	C	 Post-order traversal will visit left, right, and then node.
35.	C	^ denotes line beginning in java's regular expressions
36.	B	only 7 unique integers will be added to the hash table, so size will be 7. HashCode only affects the bucket that an element ends up in. After being bucketed, .equals() determines if a number is unique.
37.	C	In the worst case, every element added to a hash table could end up in the same bucket, making contains() a linear time method. Because add() calls contains(), add's worst case performance is O(N) as well
38.	E	The structure is a hash table that uses a linkedlist to handle collisions
39.	17	The number must be between 15 and 20, inclusive The number must be odd The number must not be a multiple of 5 The number must not end in 9 The only number to satisfy all constraints is 17
40.	12	A skinny tree can have a single node per level, for a total of 12 levels