UIL COMPUTER SCIENCE WRITTEN TEST

2022 INVITATIONAL B

FEBRUARY/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 INVITATIONAL B

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 hexadecimal numbers is equivalent to 21102_4?

A) 216_{16} **B)** 209_{16} **C)** 220_{16} **D)** 252_{16} **E)** none of these

Question 2

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

A) 0 **B)** 0.2 **C)** 73 **D)** 72 **E)** 72.2

```
out.print(9 * 8 + 7 % 6 / 5);
```

Question 3

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

A) `02.35`
B) `002.34`
C) `002.35`
D) `00002.35`
E) None of the above

```
out.printf("%05.2f", 2.345);
```

Question 4

What is the output of the line of code shown on the right?

A) 4
B) 5
C) 6
D) 7
E) 8

```
String str = "hello world";
out.print(str.lastIndexOf('o'));
```

Question 5

What is the output of the line of code shown on the right?

A) `true`
B) `false`

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

Question 6

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

A) 2.5 **B)** 3.0 **C)** 2.0 **D)** `2` **E)** 3

```
out.print(Math.round(2.45));
```

Question 7

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

A) `13.0` **B)** `17.0` **C)** `18.0` **D)** 19.0 **E)** 20.0

```
int x = 6, y = 10;
double a = 1.5, b = 1.25;
out.print(x * a + y / b);
```

Question 8

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

A) `fizz`
B) `fizzbuzz`
C) `fizzbuzz!`
D) There is no output due to a runtime error.
E) There is no output due to a compile error.

```
switch("test"){
   case "test":
      out.print("fizz");
   case "Test":
      out.print("buzz");
      break;
   case "TEST":
      out.print("!");
}
```

Question 9

How many stars are output by the code to the right?

A) 8

B) 6

C) 5

D) 7

E) 1

```
String str = "";
for(int i = 0;i<3;i++)
{
   str = str + "*";
   out.print(str);
}
```

Question 10

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

A) [6, 5, 1, 3, 1, 1]

B) [6, 5, 2, 3, 2, 1]

C) [6, 5, 2, 3, 2, 1]

D) [6, 5, 2, 3, 4, 1]

E) There is no output due to an error.

```
int[] list = {6,5,4,3,2,1};
list[list[2]] = list[list[1]];
list[2] = list[4];
out.print(Arrays.toString(list));
```

Question 11

The contents of input.txt are as follows:

```
9
10
11
12
```

What is output by the code segment to the right?

A) 10

B) 20

C) 21

D) 22

E) There is no output due to an error

```
Scanner scan;
scan = new Scanner(new File("input.txt"));
int sum = 0;
for(int i = 0;i<2;i++)
{
   scan.nextLine();
   sum += scan.nextInt();
}
out.print(sum);
```

Question 12

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

A) 18

B) 15

C) 24

D) 16

E) 10

```
int a = 1;
int h = 0;
for(int g = 1; g <= 5; g++) {
   h += a;
   a++;
}
out.print(h);
```

Question 13

What is the output of the code segment shown on the right?

A) 4

B) 3

C) 2

D) 14

E) 16

```
int a = 4, b = 2, c = 1;
out.print(a++ + ++b >> c);
```

Question 14

What is the output of the code segment shown on the right?

A) 4

B) 8

C) 16

D) 32

E) 64

```
out.print(Float.SIZE);
```

Question 15

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

A) [7, 5, 3, 1]

B) [1]

C) [7, 5, 3]

D) [7, 5, 1]

E) There is no output due to a runtime error

```
ArrayList<Integer> list;
list = new ArrayList<Integer>();
list.add(7);
list.add(5);
list.add(3);
list.add(1);
list.remove(3);
System.out.println(list);
```

Question 16

How many assignments make the following boolean expression result to true?

A) 0 **B)** 2 **C)** 4 **D)** 6 **E)** 7

(A * B) ⊕ (C * D)

Question 17

What is output by the following client code?

```
out.print(f1(10) + f2(10));
```

A) 68

B) 34

C) 110

D) 55

E) 5555

Question 18

What is the most restrictive Big O upper bound of `f1(n)`?

A) O(n)

B) O(nlogn)

C) $O(n^2)$

D) $O(2^n)$

E) O(1)

Question 19

What is the most restrictive Big O upper bound of `f2(n)`?

A) O(n)

B) O(nlogn)

C) $O(n^2)$

D) $O(2^n)$

E) O(1)

```
public static int f1(int x){
   if(x < 2)
      return x;
   return f1(x-1) + f1(x-2);
}

public static int f2(int x)
{
   int[] f = new int[x+1];
   f[0] = 0;
   f[1] = 1;
   for(int i = 2;i<f.length;i++)
      f[i] = f[i-1] + f[i-2];
   return f[x];
}
```

Question 20

How many asterisks are printed by M2(16)?

A) 60

B) 64

C) 80

D) 256

E) none of the above

```
public void M2(int n){
   for(int i = 0;i<n;i++)
   {
      for(int j = 0;j<=n;j+=4)
      {
         out.print("*");
      }
   }
}
```

Question 21

Which Java keyword is used to denote that one class is a subtype of another?

A) `super`

B) `this`

C) `parent`

D) `extends`

E) `none of the above`

Question 22

Which of the following could be generated by the following line of code?

```
out.print((int)(Math.random() * 10) * 2);
```

`I. 9  II. 10  III. 11  IV. 12`

A) `I` and `II`

B) `II` and `III`

C) `II` and `IV`

D) `II, III,` and `IV`

Question 23

What is the output of the code segment shown on the right?

A) 0

B) 5

C) 9

D) 17

E) None of the above

```
out.print(5 & 3 << 1 ^ 2);
```

Question 24

What is printed by the line labeled **<Client Code 1>** shown on the right?

A) `tob`

B) `comet`

C) `Tobor`

D) `null`

E) The code does not compile.

Question 25

What is printed by the line labeled **<Client Code 2>** shown on the right?

A) `tob`

B) `comet`

C) `Tobor`

D) `null`

E) The code does not compile.

```
TreeSet<String> set;
set = new TreeSet<String>();
set.add("tobor");
set.add("Tobor");
set.add("tob");
set.add("tobort");
//<Client Code 1>
out.print(set.floor("comet"));
//<Client Code 2>
out.print(set.higher("Tobor"));
```

Question 26

Which of the following is closest to the expected output of the code to the right?

A) `500000.0`

B) `0.5`

C) `0.0`

D) `1.0`

E) `-1.0`

```
int it = 1000000;
double sum = 0;
for(int i = 0;i<it;i++)
{
   if(Math.random() < .25)
      sum--;
   else
      sum++;
}
out.println(sum/it);
```

Question 27

What is output by the following client code?

```
DataStructure<Integer> ds;
ds = new DataStructure<>();
for(int i: new int[] {1,5,2,7,4,8,9})
   ds.add(i);
while(!ds.isEmpty())
   out.print(ds.remove()+" ");
```

A) 1 5 2 7 4 8 9

B) 9 8 4 7 2 5 1

C) 1 2 4 5 7 8 9

D) 9 8 7 5 4 2 1

E) There is no output due to a runtime error

Question 28

What is output by the following client code?

```
DataStructure<Integer> ds;
ds = new DataStructure<>();
for(int i: new int[] {1,5,2,7,4,8,9})
   ds.add(i);
while(!ds.isEmpty())
   ds.remove();
out.print(ds.next());
```

A) 0

B) o

C) null

D) 1

E) There is no output due to a runtime error

Question 29

What data structure does DataStructure represent?

A) Queue

B) PriorityQueue

C) Stack

D) Set

E) none of the above

Question 30

The O class is an example of what?

A) recursion

B) encapsulation

C) enumeration

D) inheritance

E) none of the above

```
public class DataStructure<E>{
   private class O{
      E e;
      O o;
      public O(E e, O o)
      {
         this.e = e;
         this.o = o;
      }
   }

   O o;
   O t;
   int sz;

   public void add(E e)
   {
      if(o == null) {
         o = t = new O(e,null);
      }
      else{
         t = t.o = new O(e, null);
      }
      sz++;
   }

   public E next()
   {
      return o.e;
   }

   public E remove()
   {
      E e = next();
      o = o.o;
      sz--;
      return e;
   }

   public boolean isEmpty() {
      return sz == 0;
   }
}
```

Question 31

Which of the following is the output of the line shown below?

```
out.println(Integer.MIN_VALUE - Integer.MAX_VALUE)
```

A) -1

B) 0

C) 1

D) 2

E) There is no ouput due to a runtime error

Question 32

What is output by the code shown on the right?

A) [2, 1, 2]

B) [3, 2, 3]

C) [9, 8, 7]

D) [7, 8, 9]

E) [0, 0, 0]

```
int N = 3;
int[][] ints = new int[N][N];
int[][] d;
d = new int[][] {{1,0,-1,0},{0,1,0,-1}};
Stack<Integer> st = new Stack<Integer>();
st.add(1);
st.add(1);
st.add(1);
while(!st.isEmpty())
{
   int a = st.pop();
   int b = st.pop();
   int c = st.pop();
   if(Math.min(a, b) >= 0
      && Math.max(a, b) < N
      && ints[a][b] == 0){
ints[a][b] = c;
for(int i = 0;i<4;i++)
{
   st.add(c+1);
   st.add(b+d[1][i]);
   st.add(a+d[0][i]);
}
   }
}
out.println(Arrays.toString(ints[N-1]));
```

Question 33

How many swaps would bubble sort perform to sort the array to the right in ascending order?

A) 8

B) 9

C) 10

D) 11

E) 12

```
int[] arr = new int[] {5,4,3,1,2};
```

Question 34

How many assignments of A, B, and C will make the following circuit evaluate to true?

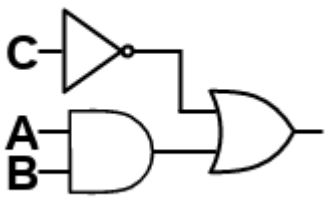
A) 1

B) 5

C) 6

D) 7

E) None of the above



Question 35

Convert the postfix expression to the right to infix

A) ((A + B) – C) * (D / E)

B) (A * (B + C) – (D / E))

C) A + B – C * D / E

D) – C + B A * / D E

E) None of the above

```
A B + C – D E / *
```

Question 36

What is the output of the code segment shown on the right?

A) []

B) [10, 11, 12, 13, 14, 15, 16, 17]

C) [10, 12, 14, 16, 18]

D) [10, 12, 14, 16]

E) There is no output due to a runtime error

```
ArrayList<Integer> list;
list = new ArrayList<Integer>();
for(int i = 0;i<18;i += 2)
list.add(i);
for(int i =0;i<9;i+=2)
list.remove(Integer.valueOf(i));
System.out.println(list);
```

Question 37

What is the output of the code to the right?

A) 000010010010

B) 010010011101

C) 010010000100

D) 000010101110

E) None of the above

```
boolean[] b = new boolean[12];
for(int i = 1;i<b.length;i++)
{
   for(int j = i;j<b.length;j+=i)
      b[j] = !b[j];
}
for(int i = 0;i<b.length;i++)
   if(b[i])
out.print(1);
   else
out.print(0);
```

Question 38

What is the output of the code to the right?

A) 17

B) 30

C) 31

D) 0

E) None of the above

```
int p = 17;
boolean flag = (p&1) == 0;
out.print(flag ? (p | 14) : (p & 14));
```

Question 39

An articulation point is a graph vertex who's removal would cause a graph to become disconnected. To the right is the adjacency matrix of graph G where adj[I][J] = adj[J][I] = 1 if a bidirectional edge exists between vertices I and J. How many articulation points exist in G?

Adj:

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

Question 40

Write the signed 8-bit binary two's complement representation of the largest 1-byte integer possible.

★ ANSWER KEY – CONFIDENTIAL ★

UIL COMPUTER SCIENCE – 2022 INVITATIONAL B

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

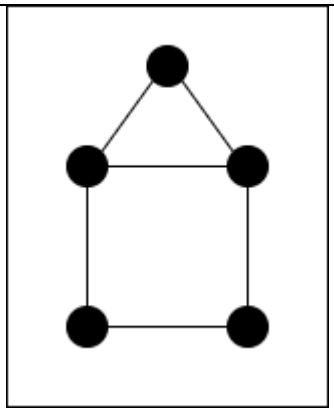
1) D	11) C	21) D	31) C
2) D	12) B	22) C	32) C
3) A	13) B	23) E	33) B
4) D	14) D	24) C	34) B
5) B	15) C	25) A	35) A
6) D	16) D	26) B	36) D
7) B	17) C	27) A	37) C
8) B	18) D	28) E	38) D
9) B	19) A	29) A	*39) 0
10) A	20) C	30) B	*40) 01111111

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

Note: Correct responses are based on **Java SE Development Kit 17 (JDK 17)** 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 17 Standard Packages have been imported. Ignore any typographical errors and assume any undefined variables are defined as used.

Explanations:

1.	D	Because 16 = 4^2, each digit two digits can be compressed into one. (2)(11)(02) = 252
2.	D	Order of operations is (9*8) + (7 % 6 / 5)
3.	A	0 pads with zeros instead of space, 5 ensures that at least 5 characters are printed, and .2 rounds to 2 decimal places
4.	D	The last o appears at index 7, (don't forget to count the space)
5.	B	^ is one or the other but not both. When chaining xor of trues and falses, an odd number of trues will result in true, and an even number of trues will result in false
6.	D	2.45 rounds down to 2
7.	B	6*1.5 = 9.0, 10/1.25 = 8.0, 9.0 + 8.0 = 17.0
8.	B	The first case executes, and then drops to the second case since there is no break.
9.	B	1 star + 2 star + 3 star = 6 star
10.	A	Initial: [6,5,4,3,2,1] After first change: [6,5,4,3,1,1] After second change: [6,5,1,3,1,1]
11.	C	The second nextLine picks up the carriage return after 10, so the numbers summed are 10 and 11.
12.	B	1 + 2 + 3 + 4 + 5 = 15
13.	B	Addition comes before shifts, so (4 + 3) / 2
14.	D	Floats are comprised of 32 bits
15.	C	List.remove(int) removes an index rather than an occurrence
16.	D	There is only 1 way to make one side of the xor true, and 3 ways to make one side of the xor false. (#true)*(#false) + (#false)*(#true) = (1)(3)+(3)(1) = 6
17.	C	F(n) calculates Fibonacci. F(10) = 55, so answer is 110
18.	D	Naïve Fibonacci is O(2^n) because each subproblem spawns 2 new ones
19.	A	Dynamic programming Fibonacci is o(n) as each subproblem is answered exactly once.
20.	C	Outer loop executes 16 times, inner loop executes 5 times, so answer is 16*5 = 80
21.	D	extends is the word used to denote subtypes
22.	C	The code will produce an even number since casting happens before multiplication
23.	E	Order of operations is shift, and, xor. 3<< 1 = 6, 5 & 6 = 4, 4 ^ 2 = 6. Answer is 6, so none of the above.
24.	C	The largest thing lexicographically smaller than comet in set is "Tobor"
25.	A	The smallest thing lexicographically larger than Tobor is "tob"
26.	B	(-1)(.25) + (1)(.75) = .5
27.	A	DataStructure represents a Queue, so trace FIFO queue operations to produce answer A
28.	E	If the structure is empty, o will be null and o.e will cause a null pointer exception.
29.	A	Datastructure is FIFO, so it implements a queue
30.	B	O encapsulates the data of an element and it's next pointer
31.	C	Max value + 1 = min value, so min value – max value = 1
32.	C	Simulate DFS with move precedence, left, up, right, down
33.	B	The number of inversions is 9, count the number of elements x, y such that x > y and x is before y in the array
34.	B	Either C is false (4 ways), or C is true and AB are both true(1 way). (5 ways total)
35.	A	Only A,B,C are valid infix expressions, and B and C both don't evaluate in the same order as the given expression. A is correct.
36.	D	List initially has every even from 0 to 16, but the second loop removes every even from 0 to 8, leaving [10,12,14,16]
37.	C	Every index in the array is flipped once for each factor of that index. Only square numbers have an odd number of factors, so non square indices will end as zeros and square numbers as ones.
38.	D	Flag is false, so the second half of the ternary is evaluated. 17 & 14 = 0

39.	0	 Since every node is part of a cycle, no single removal could disconnect the graph.
40.	01111111	127 is the largest signed 8-bit number, represented by 01111111