UIL COMPUTER SCIENCE WRITTEN TEST

2020 INVITATIONAL B

FEBRUARY/MARCH 2020

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 – 2020 INVITATIONAL B

Note: Correct responses are based on **Java SE Development Kit 12 (JDK 12)** 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 12 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 11001011_2?

A) 68_{16} **B)** CB_{16} **C)** CC_{16} **D)** $A5_{16}$ **E)** 20_{16}

Question 2.

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

A) 0 **B)** 14 **C)** 2 **D)** 9 **E)** 8

```
out.print(13 % 5 + 6 % 7);
```

Question 3.

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

A) `Invitational\\B`
B) `Invitational`
`B`
C) `InvitationalB`
D) `Invitational B`
E) `Invitational\B`

```
out.println("Invitational\\B");
```

Question 4.

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

A) `TeamGo` **B)** `GoTeam` **C)** `oTeamG`
D) `oTeamGo` **E)** `GoTeamG`

```
String str = "GoTeamGo";
out.print(str.substring(1, 7));
```

Question 5.

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

A) `true`
B) `false`

```
boolean a = true;
boolean b = true;
boolean c = a ^ b;
out.print(!c);
```

Question 6.

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

A) `3.0` **B)** `3` **C)** `1.25` **D)** `1`
E) There is no output due to an error.

```
int i = 3;
double j = 1.25;
out.print(Math.max(i, j));
```

Question 7.

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

A) `15.0` **B)** `-1` **C)** `15` **D)** `1.0` **E)** `-1.0`

```
int i = 14, j = 8;
double d = 4.0, e = 11.0;
out.print(i + d - e - j);
```

Question 8.

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

A) `7`
B) `1`
C) `48`
D) `1.1`
E) `36`

```
int r = 19, s = 12, t = 21;
if(r % s > t)
 out.print(r - s);
else if(s - t < r)
 out.print(t / r);
else
 out.print(s * 4);
```

Question 9.

Which of the following represents the output of the code shown to the right?

A) `#########`

B) `#####`

C) `########`

D) `######`

E) `#######`

```
int stop = 100;
int go = 1;
do {
 out.print("#");
 go *= 2;
}while(go < stop);
```

Question 10.

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

A) `[3, 8, -4, 5, 0]`

B) `[3, 8, -4, 5]`

C) `[0, 3, -4, -4, 5]`

D) `[0, 3, 8, -4, 5]`

E) `[0, 8, -4, 8, 5]`

```
int []nums = new int[5];
nums[1] = 3; nums[3] = -4;
nums[2] = 8; nums[4] = 5;
nums[nums[1]] = nums[nums.length-2];
out.print(Arrays.toString(nums));
```

Question 11.

Which of the following is the output of the code segment shown on the right? You may assume that all necessary import statements are present and correct. The file named `data.dat` contains the following:

```
one,two,three,four,five,six
```

A) one,two,three,four,five,six

B) one, two, three, four, five, six

C) onetwothreefourfivesix

D) one two three four five six

E) one

```
Scanner f = new Scanner(new
File("data.dat"));
f.useDelimiter(",");
while(f.hasNext())
 out.print(f.next());
f.close();
```

Question 12.

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

A) `255`

B) `256`

C) `128`

D) `12`

E) `6`

```
int m = 2, n = 0, p = 1;
while(n < 7) {
 p = p * m;
 n++;
}
out.print(p);
```

Question 13.

What is the output of the code segment shown here?

```
double x = 1.5, y = 2.25, z = -0.75;
out.print(x + z > y || z - Math.abs(x - y) > z);
```

A) `true`

B) `false`

Question 14.

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

A) 2147483647

B) 32767

C) 127

D) 2147483648

E) 32768

```
out.println(Integer.MAX_VALUE);
```

Question 15.

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

A) [E, B, D, A]

B) [E, C, D, A]

C) [E, C, B, D, A]

D) This code segment will not compile.

E) This code segment throws an exception.

```
ArrayList<String> list = new
ArrayList<String>();
list.add("E");list.add("C");list.add("B");
list.add("D");list.add("A");
list.remove(2);list.remove("F");
out.print(list);
```

Question 16.

Which of the following must replace **<code>** in the class shown on the right so that the values passed in parameters x and y are assigned to the instance variables x and y?

A) private

B) super

C) public

D) this

E) static

Question 17.

If **<code>** has been filled in correctly, which of the following lines of client code will NOT compile and execute correctly?

```
Uil uil = new Uil(5,4);
int a = Uil.y;      //line #1
int b = uil.y;      //line #2
int c = uil.getZ(); //line #3
int d = Uil.getZ(); //line #4
```

A) line #1

B) line #2

C) line #3

D) line #4

E) More than one of the above.

```
public class Uil
{
      private int x;
      public int y;
      public static int z;

      public Uil(int x, int y) {
            <code>.x = x;
            <code>.y = y;
            z += 2;
      }

      public int getX() {
            return x;
      }

      public static int getZ() {
            return z;
      }
}
```

Question 18.

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

A) false false false 0

B) true true true false

C) false false true 0

D) false true true true

E) false true true 0

```
String r = "apple";
String p = new String("apple");
String o = "apple";
String s = p;
out.print((r == p) + " ");
out.print((r == o) + " ");
out.print((r.equals(p))+" ");
out.print(p.compareTo(s));
```

Question 19.

Consider the class `FunWithNumbers` shown on the right. Which of the following must replace **<code>** to ensure that when executed the main method will produce this output?

```
[TEN, TWO, TEN, FOUR, TEN, SIX, TEN, EIGHT, TEN, TEN]
```

A) `Numbers.values(TEN)`

B) `TEN`

C) `Numbers.TEN`

D) `new Numbers(TEN)`

E) `10`

```
import static java.lang.System.out;
import java.util.Arrays;
public class FunWithNumbers {

  public enum Numbers{
    ONE,TWO,THREE,FOUR,
    FIVE,SIX,SEVEN,EIGHT,
    NINE,TEN
  }

  public static void main(String[] args) {
    Numbers[] nums = Numbers.values();
    for(int i = 0; i < nums.length; i+=2)
      nums[i] = <code>;
    out.print(Arrays.toString(nums));
  }
}
```

Question 20.

The formula to find the area of a triangle when two sides and the included angle are known is:

$$\frac{1}{2}ab\sin C$$

Which of the following statements will calculate the area of a triangle with sides a and b and an included angle c measured in degrees and assign that value to a variable named area.

A) `double area = 1 / 2 * a * b * Math.sin(Math.toRadians(c));`

B) `double area = 0.5 * a * b * Math.sin(c);`

C) `double area = 1.0 / 2.0 * ab(Math.sin(Math.toRadians(c)));`

D) `double area = 1.0 / 2 * a * b * Math.sin(toRadians(c));`

E) `double area = 0.5 * a * b * Math.sin(Math.toRadians(c));`

Question 21.

```
public static void main(String[] args) {
    <code> item = r -> r.substring(r.length() / 2);
    out.print(item.getIt("today"));
    }

public interface UILInterface {
    public String getIt(String s);
    }
```

The code segment shown above should print the second half of the string passed to the `getIt` method. What must replace **<code>** to ensure that the segment will compile and function as intended?

A) `int`

B) `public`

C) `UILInterface`

D) `String`

E) No additional code is required.

Question 22.

Which of the following best describes the method shown here?

```
public static int method(String[]list,String t) {
  int i = 0, j = -1;
  while(i < list.length) {
    if(list[i].equals(t))
      j = i;
    i++;
  }
  return j;
}
```

A) `method` returns the index value of each occurrence of `t` in `list` or 0 if `t` is not found.

B) `method` returns the index value of the first occurrence of `t` in `list` or -1 if `t` is not found.

C) `method` returns the String at index value `t` or -1 if `t` is out of bounds.

D) `method` returns the index value of the last occurrence of `t` in `list` or -1 if `t` is not found.

E) `method` always returns -1.

Question 23.

Which of the following shows the output of this client code?

```
InvB list = new InvB();
list.cat("mercury");list.cat("mars");
list.cat("earth");list.cat("jupiter");
list.bird();list.dog();list.bird();
while(!list.snake())
     out.print(list.dog() + " ");
```

A) `earth mars mercury`

B) `mars earth jupiter`

C) `mercury`

D) `jupiter earth mars`

E) `jupiter`

Question 24.

The class InvB implements a _____.

A) queue

B) map

C) stack

D) set

E) priority queue

```
//Use the class shown here to answer
// questions 23 and 24.
import java.util.*;
public class InvB {

  private ArrayList<String> list;

  public InvB() {
    list = new ArrayList<String>();
  }

  public void cat(String s) {
    list.add(s);
  }

  public String dog() {
    return list.remove(list.size() - 1);
  }

  public String bird() {
    return list.get(list.size() - 1);
  }

  public boolean snake() {
    return list.size() == 0;
  }
}
```

Question 25.

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

A) `five 4`

B) `zero 4`

C) `one 5`

D) `three 5`

E) `two 6`

```
Queue<String> list = new
LinkedList<String>();
list.add("two");list.add("one");
list.add("five");list.add("zero");
list.add("three");list.add("four");
list.poll();list.remove();
out.print(list.peek() + " " + list.size());
```

Question 26.

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

A) * * * # # 1 -1 3

B) * * * * * 2 -2 3

C) * * * * * -1 1 3

D) * * * # # 2 -2 3

E) # # # * * -1 1 3

```
int x = 4, y = -4, a = -2;
while(a <= 2) {
  if(a++ > 0 && x-- == ++y)
    out.print("# ");
  else
    out.print("* ");
  }
out.print(x + " " + y + " " + a);
```

Question 27.

Which of the following must replace **<code>** to ensure that the method will compile and execute as intended?

A) `k<=0 && list[k]<current`

B) `k>=0 || list[k]>current`

C) `k>=0 && list[k]>current`

D) `list[k]>current`

E) `k<=0`

Question 28.

Assume that **<code>** has been replaced with the correct code and this client code is executed.

```
int[] list = {6,7,1,8,2,9,0,5,4,3};
insertion(list);
```

What is printed when `i` equals 4 if this line of code

```
out.println(Arrays.toString(list));
```

replaces the comment?

A) `[1, 2, 0, 5, 4, 3, 6, 7, 8, 9]`

B) `[0, 1, 2, 3, 4, 9, 6, 5, 7, 8]`

C) `[1, 6, 7, 8, 2, 9, 0, 5, 4, 3]`

D) `[1, 2, 6, 7, 8, 9, 0, 5, 4, 3]`

E) `[0, 1, 2, 3, 7, 9, 6, 5, 4, 8]`

Question 29.

Once the method `insertion` has been correctly implemented, what is the worst case time complexity for this method?

A) O(1)

B) O(n)

C) O(log n)

D) O(n log n)

E) $O(n^2)$

```
//Use the following implementation of
//an insertion sort to answer
//questions 27, 28 and 29.

public static void insertion(int[] list)
{
 for(int i = 1; i < list.length;i++)
 {
  int current = list[i];
  int k;
  for(k = i - 1; <code>; k--)
    list[k + 1] = list[k];
  list[k + 1] = current;
  //comment
 }
}
```

Question 30.

Which of the following represents the output of the code segment shown on the right?

A)
```
[1, 4, 7]
[2, 5, 8]
[3, 6, 9]
```

B)
```
[7, 8, 9]
[4, 5, 6]
[1, 2, 3]
```

C)
```
[9, 6, 3]
[8, 5, 2]
[7, 4, 1]
```

D)
```
[3, 2, 1]
[6, 5, 4]
[9, 8, 7]
```

E)
```
[3, 6, 9]
[2, 5, 8]
[1, 4, 7]
```

```
int[][] mat = {{1,2,3},{4,5,6},{7,8,9}};
int len = mat.length - 1;
for(int r = 0; r < mat.length; r++) {
  for( int c = 0; c < len; c++) {
    int t = mat[r][c];
    mat[r][c]=mat[mat.length - 1 - c][len];
    mat[mat.length - 1 - c][len] = t;
    }
  len--;
  }
for(int[] a:mat)
  out.println(Arrays.toString(a));
```

Question 31.

Which of the following represents the output of the line of code shown on the right?

A) 9

B) 27

C) 1

D) 24

E) 0

```
System.out.print(3<<3);
```

Question 32.

If a particular method whose run time efficiency is $O(n^2)$ requires 1 second to process 12000 elements in a data set, how long will it take to process 48000 elements?

A) 2 seconds **B)** 4 seconds **C)** 8 seconds **D)** 16 seconds **E)** 32 seconds

Question 33.

Which of the following is NOT a valid identifier?

A) `amount` **B)** `$Dollar` **C)** `%percent` **D)** `last` **E)** `three_4`

Question 34.

Which of the following is the correct method header for a method that returns a tip amount, given the amount of the check and the desired percent tip?

A) `public static tip(double check, double percent)`

B) `public static double tip(double check, double percent)`

C) `public static int tip(double check, double percent)`

D) `public static double tip(check, percent)`

E) `public static double (double check, double percent)`

Question 35.

Which of the following is the truth table for this expression? $\bar{A} + B \oplus A$

A.

A	B	
T	T	T
T	F	T
F	T	T
F	F	T

B.

A	B	
T	T	T
T	F	F
F	T	T
F	F	F

C.

A	B	
T	T	F
T	F	T
F	T	T
F	F	F

D.

A	B	
T	T	T
T	F	T
F	T	T
F	F	F

E.

A	B	
T	T	F
T	F	T
F	T	T
F	F	T

Question 36.

Which of the following represents output of the main method shown on the right?

A) `*#######r`

B) `r#######*`

C) `*#######c`

D) `*r#######`

E) `r*#######`

```
public static void main(String[] args)
{
  out.print(rec("computer"));
}

public static String rec(String x)
{
  if(x.length()>1)
    {
    return "#" + rec(x.substring(1));
    }
  else
    {
    out.print("*");
    return x;
    }
}
```

Question 37.

What is the value of the postfix expression shown on the right? The operands are 8, 3, 2, 14, and 5.

A) 37

B) 7

C) 29

D) 11

E) 22

```
8 3 * 2 + 14 5 + -
```

Question 38.

Which of the following Boolean expressions is diagrammed on the right?

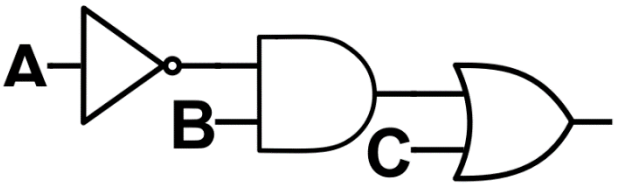
A) `A && !B || C`

B) `!A || B && C`

C) `!(A && B || C)`

D) `A && B || C`

E) `!A && B || C`



Question 39.

Determine the output of the code segment shown on the right and write your answer in the blank provided on the answer document.

```
char[] c = {'a','m','e','x','r','c'};
int h = 200;
for(char ch:c)
  switch(ch) {
    case 'a':h-='a';break;
    case 'm':h+='m';break;
    case 'x':h-='x';
    case 'c':h+='c';break;
    default: h++;
  }
out.print(h);
```

Question 40.

If the values shown here are placed into a binary search tree in the order that they are listed, which value will serve as the root node? Write your answer in the blank provided on the answer document.

2 0 8 4 6 7 1 3 5 9