UIL COMPUTER SCIENCE WRITTEN TEST

2017 INVITATIONAL B

FEBRUARY/MARCH 2017

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)
```

UIL COMPUTER SCIENCE WRITTEN TEST – 2017 INVITATIONAL B

Note: Correct responses are based on **Java SE Development Kit 8 (JDK 8)** 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 8 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 the sum of 01101001_2 and 00101011_2?

A) 00101000_2 **B)** 10010101_2 **C)** 10011100_2 **D)** 10010111_2 **E)** 10010100_2

Question 2.

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

A) 8 **B)** 0.125 **C)** 2 **D)** 0 **E)** 2.0

```
out.println((9+2)/22*4);
```

Question 3.

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

A)
```
Hello"r"World!
```
B)
```
Hello"
"World!
```
C)
```
Hello
World!
```
D)
```
Hello World!
```
E)
```
"Hello"r"World!"
```

```
out.print("Hello\"\r\"World!");
```

Question 4.

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

A) crossc **B)** rossc **C)** cross

D) ross **E)** Error. Cannot call substring method using a string constant.

```
out.print("crosscountry".substring(1, 5));
```

Question 5.

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

A) `true` **B)** `false`

```
out.print(true&&!false&&true);
```

Question 6.

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

A) `-9.0` **B)** `-9` **C)** `9.0` **D)** `9` **E)** `3.0`

```
out.print(Math.abs(9.0));
```

Question 7.

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

A) `-9.0` **B)** `5.25` **C)** `-5.25` **D)** `-9.75`

E) Error. Will not compile because of a type mismatch.

```
double p=4.5;
int q=3;
double r=-2.25;
r-=p+q;
out.print(r);
```

Question 8.

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

A) i U L

B) U i

C) i U

D) U i L

E) L

```
char c1='U',c2='i',c3='L';
if(c1<c2)
   out.print(c1+" "+c2+" ");
else
   out.print(c2+" "+c1+" ");
out.println(c3);
```

Question 9.

Which of the following represents the output of this code segment?

```
for(int x=4;x>0;x--)
     {
     for(int y=1;y<=x;y++)
          out.print("*");
     out.println();
     }
```

A.
```
****
***
**
*
```

B.
```
*
**
***
****
```

C.
```
****
****
****
****
```

D.
```
**********
```

E.
```
*
```

Question 10.

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

A) 8 1 -3 1 2

B) 5 -1 5 1 2

C) 8 1 8 1 2

D) -1 1 -3 1 2

E) -1 1 1 7 2

```
int[] a={5,1,-3,7,2};
a[0]=a[2]+a[4];
a[3]=a[1];
for(int x:a)
  out.print(x+" ");
```

Question 11.

Which of the following import statements is required to make this class compile and run correctly?

```
public class Abc {

     public static void main(String[] args) throws IOException{
          Scanner s=new Scanner(new File("datafile.dat"));
          while(s.hasNext())
               out.println(s.nextLine());
     }
}
```

A) import static java.lang.System.out;

B) import java.io.File;

C) import java.io.IOException;

D) import java.util.Scanner;

E) All of the above.

Question 12.

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

A) 9

B) 5

C) 6

D) 8

E) 1

```
int w=1,x=2,y=3;
for(int z=1;z<=y;z++)
     w*=x;
out.print(w);
```

Question 13.

What is the correct order of operation for the operators listed here?

I. = II. ++ III. && IV. &

A) III, IV, I, II

B) IV, III, II, I

C) II, I, IV, III

D) II, III, IV, I

E) II, IV, III, I

Question 14.

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

A) 8 **B)** 16 **C)** 32 **D)** 64 **E)** None of the above.

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

Question 15.

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

A) [dog, cat]

B) [dog, bird, cat]

C) [dog, cat, bird]

D) [dog, bird, turtle]

E) Error. ElementNotFoundException.

```
ArrayList<String> a=new
ArrayList<String>();
a.add("dog");
a.add("cat");
a.add(1,"bird");
a.add("turtle");
a.remove(3);
a.remove("turtle");
out.print(a);
```

Question 16.

Which of the following can replace **<code>** in the code segment shown to the right?

A) ArrayList

B) List

C) LinkedList

D) A and C

E) B and C

Question 17.

Assuming that **<code>** has been filled in correctly, what is the output of the code segment to the right?

A) 15 [30, 5, 10, 35, 15, 20, 25]

B) 15 [5, 10, 35, 15, 20, 25, 30]

C) 20 [5, 10, 35, 15, 20, 25, 30]

D) 20 [5, 10, 35, 15, 25, 30]

E) 15 [30, 5, 10, 35, 20, 25]

```
<code><Integer> list=new
LinkedList<Integer>();
int[] nums={5,10,15,20,25};
for(int i:nums)
     list.add(i);
list.add(30);
list.add(2, 35);
out.print(list.get(4)+" ");
out.print(list);
```

Question 18.

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

A) x y z mno

B) abc

C) xaybzcmno

D) xyzmno

E) None of the above.

```
String s="xaybzcmno";
Scanner s2=new Scanner(s);
s2.useDelimiter("[abc]");
while(s2.hasNext())
     out.print(s2.next());
```

Question 19.

If a particular method whose run time efficiency is $O(n^2)$ requires 1 second to process 2000 elements in a data set, how long will it take to process 10000 elements?

A) 25 seconds **B)** 20 seconds **C)** 10 seconds **D)** 8 seconds **E)** 64 seconds

Question 20.

Which of the following is not a valid identifier?

A) _mileage **B)** $amount **C)** 7thChar **D)** firstName **E)** print

Question 21.

Which of the following must replace **<code>** in the method shown to the right so that the values stored in list are placed in ascending order?

A) `list[i]<s`

B) `list[i]<x`

C) `list[i]<y`

D) `list[x]<y`

E) `list[y]<x`

Question 22.

Once **<code>** has been replaced correctly, which of the following algorithms does method abc implement?

A) selection sort

B) insertion sort

C) mergesort

D) quicksort

E) bubble sort

Question 23.

What is the Big O value for method abc?

A) O(1)

B) O(n log n)

C) O(log n)

D) O(n)

E) $O(n^2)$

```
// use method abc to answer questions 21,
// 22, and 23

public static void abc(int[] list){
for(int s=0;s<list.length;s++){
      int x=s;
      int y=list[s];
      for(int i=s+1;i<list.length;i++){
            if(<code>){
                  x = i;
                  y=list[x];
            }
      }
      list[x]=list[s];
      list[s]=y;
}
}
```

Question 24.

Which of the following must replace **<code1>** in the class shown on the right so that the values passed to parameters a and b are correctly assigned to the fields a and b?

A) `super.`

B) `this.`

C) `null.`

D) `int`

E) No additional code is needed

Question 25.

Which of the following must replace **<code2>** in the class shown on the right to ensure that the toString method functions correctly?

A) `int`

B) `out.print`

C) `return`

D) `String`

E) No additional code is needed

Question 26.

Assuming that **<code1>** and **<code2>** have been filled in correctly, what is the output of this client code?

```
SomeClass sc1=new SomeClass(3,2);
SomeClass sc2=new SomeClass();
out.print(sc1+" "+sc2);
```

A) 3 2

B) 3 0 2 0

C) 0 0 0 0

D) 3 2 `null null`

E) 3 2 0 0

```
// Use the code for SomeClass to
// answer questions 24, 25, and 26.

public class SomeClass {
     private int a;
     private int b;

     public SomeClass(){}

     public SomeClass(int a,int b){
          <code1>a=a;
          <code1>b=b;
     }

     public String toString(){
          <code2> (a+" "+b);
     }
}
```

Question 27.

Which of the following reserved words must take the place of **<code1>** to make class B inherit class A's fields, constructors and methods?

A) instanceof

B) super

C) static

D) extends

E) implements

Question 28.

The add() method in class B is intended to find the sum of fields x, y, and z. Which of the following should replace **<code2>** to ensure that add() functions as intended?

A) `return super.add(z);`

B) `return super.add()+z;`

C) `return x+y+z;`

D) `return super.x+super.y+z;`

E) B, C, and D

```
// Use for questions 27 and 28.

public class A {

     private int x;
     private int y;

     public A(){}
     public A(int a,int b){
          x=a;
          y=b;}
     public void setX(int a){
          x=a;
     }
     public void setY(int a){
          y=a;
     }
     public int add(){
          return x+y;}
}

public class B <code1> A {
     private int z;

     public B(int m,int n,int o){
          setX(m);
          setY(n);
          z=o;
          }
     public int add(){
          <code2>}
}
```

Question 29.

What is the final state of matrix x after this client code has been executed? Method abc is shown to the right.

```
int[][] x={{2,5,8},{6,5,4},{1,2,3}};
x=abc(x);
```

A) 5 8 2
5 4 6
2 3 1

B) 1 2 3
2 5 8
6 5 4

C) 3 4 8
1 5 5
1 6 2

D) Error. Method abc throws an ArrayIndexOutOfBoundsException.

E) Error. Invalid call to method abc in client code.

```
public static int[][] abc(int[][] m){
for(int r=0;r<m.length;r++){
     int t=m[r][0];
     for(int c=0;c<m[r].length-1;c++)
          m[r][c]=m[r][c+1];
     m[r][m[r].length-1]=t;
}
return m;
}
```

Question 30.

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

A) 5 31

B) 4 30

C) 6 63

D) 5 7

E) 5 30

```
int e=1,g=1;
while(e<5){
     if(g%2==0)
          g+=1;
     else{
          g*=2;
          continue;}
     e++;
}
out.print(e+" "+g);
```

Question 31.

What the output of the method shown to the right if the client code contains this method call?

```
rec(9);
```

A) 1 2 4 9

B) 0 1 2 4 9

C) 9 4 2 1 0

D) 1 2 4

E) 9 4 2 1

```
public static void rec(int x){
if(x<=1)
     out.print(x+" ");
else
     {
     rec(x/2);
     out.print(x+" ");
     }
}
```

Question 32.

Which of the following Java statements will compile and is equivalent to the mathematical formula shown to the right?

$$A = 1/2(b1 + b2)h$$

A) double a=1/2*(b1+b2)*h;

B) double a=1.0/2.0*b1+b2*h;

C) double a=1.0/2.0*h*(b1+b2);

D) All of the above.

E) None of the above.

Question 33.

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

```
double var1=8.0,var2;
final float var3=4.5f;
var2=var1+var3;
var3+=var2+var1;
out.print(var1+" "+var2+" "+var3);
```

A) 8 12.5 25

B) 8.0 12.5 25.0

C) Error. Cannot print variables declared as final.

D) Error. Type mismatch: cannot convert from double to float.

E) Error. The final local variable var3 cannot be reassigned.

Question 34.

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

```
String s1="Computer",s2="Computation";
out.print(s1.compareTo(s2));
```

A) -5

B) 5

C) -4

D) 4

E) 0

Question 35.

Which of the following is the correct method header for a method that returns a sales person's commission, given their total sales and their commission rate?

A) `public static double com(ts,cr)`

B) `public static double com(double ts,double cr)`

C) `public static com(double ts,double cr)`

D) `public static double com(double,double)`

E) `public static double (double ts,double cr)`

Question 36.

Which of the following Boolean expressions will produce the truth table shown on the right?

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

A) $A \oplus B \oplus A$

B) $A \oplus B * A$

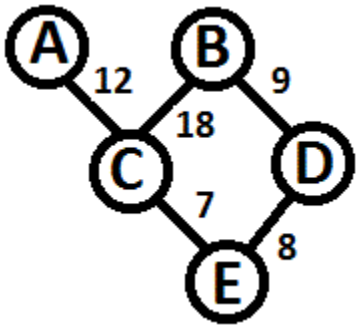
C) $A * B + A$

D) $A \oplus B + A$

E) $A * B * A$

Question 37.

Which of the following best describes the graph shown to the right?



A) unweighted directed and complete

B) unweighted directed

C) unweighted undirected

D) weighted undirected and complete

E) weighted undirected

Question 38.

What is the time complexity (Big O value) for adding an element to the end of a singly linked list that contains n elements?

A) O(1)

B) O(n)

C) $O(n^2)$

D) O(log n)

E) O(n log n)

Question 39.

What is the decimal equivalent of this 8 bit binary number which is shown in 2's complement?

10101110

Question 40.

What is the value of this expression shown using prefix notation?

+ * / 12 4 3 10