University Interscholastic League

Computer Science Competition

Number 119 (Invitational A - 2010)

General Directions (Please read carefully!):

1) DO NOT OPEN EXAM UNTIL TOLD TO DO SO.

2) **NO CALCULATOR OF ANY KIND MAY BE USED.**

3) There are 40 questions on this contest exam. You have 45 minutes to complete this contest. If you are in the process of actually writing an answer when the signal to stop is given, you may finish writing that answer.

4) Papers 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 paper until told to do otherwise. Use this time to check your answers.

5) All answers must be written on the answer sheet/Scantron card provided. Indicate your answers in the appropriate blanks provided on the answer sheet or on the Scantron card. Clean erasures are necessary for accurate Scantron grading.

6) You may place as many notations as you desire anywhere on the test paper, but not on the answer sheet or Scantron card which are reserved for answers only.

7) You may use additional scratch paper provided by the contest director.

8) All questions have ONE and only ONE correct (BEST) answer. There is a penalty for all incorrect answers. **All provided code segments are intended to be syntactically correct, unless otherwise stated. Ignore any typographical errors and assume any undefined variables are defined as used.**

9) A reference to commonly used Java classes is provided at the end of the test, and you may use this reference sheet during the contest. You may detach the reference sheets from the test booklet, but DO NOT DO SO UNTIL THE CONTEST BEGINS.

10) Assume that any necessary import statements for standard Java packages and classes (e.g. `.util`, `ArrayList`, etc.) are included in any programs or code segments that refer to methods from these classes and packages.

Scoring:

1) All questions will receive **6 points** if answered correctly; no points will be given or subtracted if unanswered; **2 points** will be deducted for an incorrect answer.

QUESTION 1

What is the sum of 1001_2 and 11_2?

A. 1100_2 B. 1111_2 C. 1010_2 D. 111_2 E. 1011_2

QUESTION 2

What is output by the code to the right?

A. 2 B. 1 C. 0

D. 6 E. 1.5

```
int x = 3000 / 10 / 100 * 2;
System.out.print(x);
```

QUESTION 3

What is output by the code to the right?

A. 19 B. 10 C. 20

D. 0 E. 1

```
int count = 0;
for(int i = 0; i < 20; i++)
  count++;
System.out.print(count);
```

QUESTION 4

What is output by the code to the right?

A. Yu B. oo C. yu

D. or E. Yourdon

```
String nm = "Yourdon";
String part = "";
part = part + nm.charAt(1) + nm.charAt(3);
System.out.println(part);
```

QUESTION 5

What is output by the code to the right?

A. 0 B. 4 C. 3

D. null E. 5

```
int[] nums = {5, 1, 7, 5, 5, 3, 5};
System.out.print(nums[5]);
```

QUESTION 6

What is output by the code to the right?

A. 1 B. 42.5 C. 5

D. 4.5 E. 5.0

```
int w = 10;
double c = w / 4 + 2.5;
System.out.print(c);
```

QUESTION 7

Which answer is logically equivalent to the following `boolean` expression, where `p` and `q` are `boolean` variables?

```
!(p || !q)
```

A. `!q` B. `!p && q` C. `!p || q` D. `!p && !q` E. `p`

QUESTION 8

What is output by the code to the right?

A. 2 B. 12 C. 1

D. 123 E. 13

```
int x = 12;
int y = 12;
if( x < y )
  System.out.print(1);
if( x <= y )
  System.out.print(2);
else
  System.out.print(3);
```

QUESTION 9

What replaces **<*1>** in the code to the right to call the constructor in the `City` class with two parameters using `nm` as the first argument and `0` as the second?

A. `super(nm, 0)`

B. `this(nm, 0)`

C. `this.City(nm, 0)`

D. `super.City(nm, 0)`

E. `City(nm, 0)`

Assume **<*1>** is filled in correctly.

```
public class City{

  private String name;
  private int pop;

  public City(String nm){
    <*1>;
  }

  public City(String nm, int p){
    name = nm;
    pop = p;
  }

  public String toString(){
    return name + " " + pop;
  }
}

////////////////////////////////////////
// client code
City c1 = new City("Waco", 100000);
System.out.println( c1 );
```

QUESTION 10

What is output by the client code to the right?

A. `waco 100,000`

B. `c1`

C. `Waco 0`

D. `WACO`

E. `Waco 100000`

QUESTION 11

What is output by the code to the right?

A. `-11 -11` B. `10 10` C. `10 -1`

D. `10 -2` E. `10 -11`

```
int bx = 10;
int ax = ~bx;
System.out.print(bx + " " + ax);
```

QUESTION 12

What is output by the code to the right?

A. `32` B. `32.0` C. `25`

D. `25.0` E. `10.0`

```
double res = Math.pow(5, 2);
System.out.print(res);
```

QUESTION 13

What is output by the code to the right?

A. `P T` B. `Pt` C. `PTT`

D. `P\tT` E. `PtT`

```
String letters = "P\tT";
System.out.print(letters);
```

QUESTION 14

What is output by the code to the right?

A. `19.5910000` B. `20.0`

C. `7.5` D. `19.5910`

E. `19.59100`

```
double value = 19.591;
System.out.printf("%7.5f", value);
```

QUESTION 15

What is returned by the method call `process(3)`?

A. `9` B. `8` C. `0`

D. `3` E. `5`

```
public int process(int x){
  int y = x;
  x++;
  y--;
  return x * y;
}
```

QUESTION 16

How many `'*'`s are output by the code to the right?

A. `20` B. `50` C. `6`

D. `15` E. `30`

```
for(int r = 0; r < 10; r++)
  for(int c = 0; c < 5; c++)
    System.out.print('*');
```

QUESTION 17

What is output by the code to the right?

A. `cara` B. `bata` C. `:-):-(`

D. `::-)` E. `batd`

```
String garbage = ":car:-)bat:-(a!!d";
String arg1 = "[^a-zA-Z]+";
String[] parts = garbage.split(arg1);
System.out.print( parts[1] + parts[3] );
```

QUESTION 18

What is output by the code to the right?

A. `4` B. `17` C. `32`

D. `34` E. `15`

```
int[] vals = {-2, 0, 7, 10, 12, 3, 2};
int total = 0;
for(int i = 0; i < vals.length; i++){
  total += vals[i];
  if(total >= 10)
    break;
}
System.out.print(total);
```

QUESTION 19

What is output by the code to the right?

A. `91` B. `17` C. `'Q'`

D. `'17'` E. `Q`

```
int offset = 3;
char ch = 'N';
ch = (char)(ch + offset);
System.out.print(ch);
```

QUESTION 20

What is output by the code to the right?

A. `ABD` B. `AAAAAAA`

C. `BCE` D. `BCDEFG`

E. There is no output due to a `StringIndexOutOfBoundsException`.

```
String start = "ABCDEFG";
String result = "";
for(int i = 1; i < start.length(); i += i)
  result += start.charAt(i);
System.out.print(result);
```

QUESTION 21

Consider the class headers to the right. Assume all of the classes to the right have a default constructor. Which of the following statements will cause a syntax error?

I. `Media m1 = new TVShow();`
II. `Media m2 = new SchoolPaper();`
III. `SchoolPaper p1 = new Paper();`

A. I only B. II only C. III only

D. I and II only E. I, II, and III

```
public class Media

public class Paper extends Media

public class SchoolPaper extends Paper

public class TVShow extends Media
```

QUESTION 22

What is output by the following client code?

```
Brick b1 = new Brick();
Brick b2 = new Brick();
b1.hit();
System.out.print( b1.isShowing() + " ");
System.out.print( b2.isShowing() );
```

A. `false true` B. `true false`

C. `false false` D. `true true`

E. There is no output due to a runtime error caused by the client code.

```
public class Brick{
  private boolean hidden;

  public void hit(){
    hidden = true;
  }

  public boolean isShowing(){
    return !hidden;
  }
}
```

QUESTION 23

This question makes use of the `Brick` class from question 22. What is output by the following client code?

```
ToughBrick t1 = new ToughBrick();
t1.hit();
System.out.print( t1.isShowing() + " ");
t1.hit();
System.out.print( t1.isShowing() );
```

A. `false true` B. `true false`

C. `false false` D. `true true`

E. There is no output due to a syntax error in the `ToughBrick` class.

```
public class ToughBrick extends Brick{
  private int reqHits;
  private int hits;

  public ToughBrick(){
    reqHits = 2;
  }

  public void hit(){
    hits++;
    if(hits == reqHits)
      hidden = true;
  }
}
```

QUESTION 24

Which of the following is not a Java keyword?

A. `throw` B. `switch` C. `do` D. `finally` E. `range`

QUESTION 25

What is output by the code to the right?

A. `3 2` B. `3 -2.6` C. `-5 -2`

D. `-3 -2` E. `3 -2`

```
int num = -13;
int div = 5;
System.out.print( num % div );
System.out.print( " " + (num / div) );
```

QUESTION 26

The quicksort algorithm sorts values in an array into ascending order based on the following algorithm:

```
pick the pivot
partition the elements of the array based on the pivot
quicksort the elements less than or equal to the pivot
quicksort the elements greater than the pivot
```

Given the following initial array:

```
{5, 8, 12, 10, -5, 7, 13}
```

Which of the following is a possible ordering of the elements of the array after the first partition assuming the middle element of the array (the value `10` in this case) is chosen as the first pivot?

A. `{-5, 8, 7, 5, 10, 12, 13}`

B. `{13, 5, 12, 10, -5, 7, 8}`

C. `{10, 5, 8, -5, 7, 12, 13}`

D. `{12, 13, 5, -5, 8, 7, 10}`

E. `{5, 8, 12, 10, -5, 7, 13}`

QUESTION 27

Given an array of N distinct elements in random order, what is the Big O of the traditional implementation of the quicksort algorithm?

A. O(NlogN) B. $O(N^2)$ C. $O(N^2 \log N)$ D. $O(N^{3/2})$ E. $O(N^3)$

QUESTION 28

What is output by the code to the right?

A. `3.0` B. `4` C. `4.0`

D. `3` E. `5`

```
double aa = 2.4;
double bb = 1.6;
int x = (int)aa + (int)bb;
System.out.print(x);
```

QUESTION 29

What is output by the code to the right?

A. `false0` B. `-3`

C. `true0` D. `4`

E. There is no output due to an `ArrayIndexOutOfBoundsException`.

```
int[] small = {-3, 4, 1, 2};
int index = small[0];
if(index > 0 && index < small.length
    && small[index] > index){
  System.out.print( small[index] );
}
System.out.print(index);
```

QUESTION 30

What is output by the code to the right?

A. `"XA"` B. `XAXAXAXA`

C. `XAXAXA` D. `XXXAAA`

E. `"XAXAXA"`

```
String st = "XA";
st = st + st + st;
System.out.print(st);
```

QUESTION 31

Method `max` to the right is suppose to return the max value in the array named `list` but it does not always work as intended. What must be changed to make the method work as intended?

A. Change the segment `if( d > max )` to `if( d <= max)`

B. Change the segment `for(double d : list)` to `for(double d : list<Double>)`

C. Change the segment `for(double d : list)` to `for(double d = list[0] : list)`

D. Change the statement `return max;` to `return d;`

E. Change the statement `double max = 0.0;` to `double max = list[0];`

```
// pre: list.length > 0
public double max(double[] list){
  double max = 0.0;
  for(double d : list)
    if( d > max )
      max = d;
  return max;
}
```

QUESTION 32

What is output by the code to the right?

A. `true[l, m, n]`

B. `true[l, m, m, n]`

C. `false[l, m, m, n]`

D. `null[l, m, n]`

E. `true[a, b, u]`

```
Set<Character> chs;
chs = new TreeSet<Character>();
String ta1 = "milner";
for(int i = 0; i < ta1.length(); i++)
  chs.add(ta1.charAt(i));

ArrayList<Character> lst;
lst = new ArrayList<Character>();
String ta2 = "manblum";
for(int i = 0; i < ta2.length(); i++)
  lst.add(ta2.charAt(i));

System.out.print(chs.retainAll(lst));
System.out.print(chs);
```

QUESTION 33

What is output by the code to the right?

A. `true` B. `25` C. `9`

D. `15` E. `false`

```
int b1 = 8;
int b2 = 7;
b2 = b1 | b2 & 17;
System.out.print(b2);
```

QUESTION 34

What is output by the code to the right?

A. `7 7` B. `10 13` C. `13 10`

D. `7 10` E. `13 7`

```
PriorityQueue<Integer> pq;
pq = new PriorityQueue<Integer>();
pq.add(13);
pq.add(7);
pq.add(7);
pq.add(10);
System.out.print( pq.remove() + " ");
System.out.print( pq.remove() );
```

QUESTION 35

What is output by the code to the right?

A. `IB`

B. `OASIB`

C. `OASJLR`

D. `GMLAIB`

E. There is no output due to a `NoSuchElementException`.

```
String[] initials = {"GM", "OAS", "LA",
                                "IB", "JLR"};
ArrayList<String> con;
con = new ArrayList<String>();
for(String s : initials)
  con.add(s);

Iterator<String> it = con.iterator();
while(it.hasNext())
  if(it.next().length() == 2)
    System.out.print(it.next());
```

QUESTION 36

Which of the following best describes what method `problem` returns?

A. The maximum value in `mat`.

B. The index of the row in `mat` closest to `0` that contains the minimum value in `mat`.

C. The index of the last row in `mat` that contains a negative value.

D. The index of the row in `mat` closest to `0` that contains the most negative values.

E. The minimum value in `mat`.

```
public int problem(double[][] mat){
  int res = 0;
  int m = -1;
  for(int i = 0; i < mat.length; i++){
    int var = 0;
    for(int j = 0; j < mat[i].length; j++)
      if( mat[i][j] < 0 )
        var++;
    if( var > m ){
      res = i;
      m = var;
    }
  }
  return res;
}
```

QUESTION 37

What is output by the code to the right?

A. `[a, a, A, B]` B. `[A, B, a, a]`

C. `[B, A, a, a]` D. `[A, a, a, B]`

E. `[a, A, B]`

```
char[] arr = {'a', 'B', 'A', 'a'};
Arrays.sort(arr);
System.out.println( Arrays.toString(arr) );
```

QUESTION 38

What is output by the code to the right?

A. `null` B. `true` C. `false`

D. There is no output due to a syntax error.

E. There is no output due to a runtime error.

```
String nm = "Wirth";
boolean result = nm instanceof ArrayList;
System.out.println(result);
```

QUESTION 39

What is output by the following client code?

```
Structure<String> s;
s = new Structure<String>();
s.add("A");
s.add("B");
s.add("B");
s.add("AB");
s.add("A");
System.out.print(s.size());
```

A. 3 B. 2 C. 5

D. 10 E. 6

QUESTION 40

What type of data structure does the `Structure` class implement?

A. A stack B. A set C. A heap

D. A queue E. A list

```
public class Structure<E>{

  private ArrayList<E> con;

  public Structure(){
    con = new ArrayList<E>();
  }

  public boolean add(E obj){
    boolean result = con.contains(obj);
    if( !result )
      con.add(obj);
    return result;
  }

  public boolean present(E obj){
    return con.contains(obj);
  }

  public int size(){
    return con.size();
  }

  public boolean remove(E obj){
    return con.remove(obj);
  }
}
```