

University Interscholastic League

Computer Science Competition

Number 130 (State - 2011)

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 517_8 and 11110101_2 ?

- A. 400_{10} B. 882_{10} C. 580_{10} D. 882_{16} E. $A4_{16}$

QUESTION 2

What is output by the code to the right?

- A. 5650 B. 226 C. 220
D. 3 E. 2

```
int x = 113;
x /= 5 * 10;
System.out.print(x);
```

QUESTION 3

What is output by the code to the right?

- A. 45 B. 35 C. 25
D. 10 E. 5

```
int val = 25;
for(int i = 0; i <= 9; i++){
    val--;
    --val;
}
System.out.print(val);
```

QUESTION 4

What is output by the code to the right?

- A. 3 3 B. 6 9 C. 3 9
D. 3 15 E. 6 30

```
String name = "Yao";
String con = name + name;
con = con + con + name;
String r;
r = name.length() + " " + con.length();
System.out.print(r);
```

QUESTION 5

What is output by the code to the right?

- A. [5, 0, 1, 2, 3, 2]
B. [6, 0, 1, 2, 3, 1]
C. [6, 2, -2, 2, 3, 1]
D. [5, 2, -2, -1, 3, 2]
E. [5, 2, 1, 2, 3, 1]

```
int[] data = {5, 2, 1, 2, 3, 1};
data[data[2]] -= data[data[4]];
data[data[0]]++;
System.out.print(Arrays.toString(data));
```

QUESTION 6

What is output by the code to the right?

- A. 0 B. 0.001 C. 0.02
D. 0.05 E. 0.25

```
double a2 = 20.25;
System.out.print(a2 % 10);
```

QUESTION 7

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

$(p \ \&\& \ q) \ || \ (p \ \&\& \ !r)$

- A. $q \ || \ !r$ B. $q \ \&\& \ !r$ C. $(q \ \&\& \ !r) \ || \ p$
D. $p \ \&\& \ (q \ || \ !r)$ E. $p \ \&\& \ (!q \ \&\& \ r)$

<p>QUESTION 8</p> <p>What is output by the code to the right?</p> <p>A. 2 27 B. 2 26 C. 2 -9</p> <p>D. 12 -10 E. 2 -10</p>	<pre>int x1 = 5; int z = -10; if(x1 > 10 && z + 1 > 0) System.out.print(1); else System.out.print(2); System.out.print(" " + z);</pre>
<p>QUESTION 9</p> <p>What replaces <*1> in the client code to the right so that the method named strong from the CheckedPassword class is called?</p> <p>A. p.strong()</p> <p>B. ((Password) p).strong()</p> <p>C. CheckedPassword.strong()</p> <p>D. extends.strong()</p> <p>E. ((CheckedPassword) p).strong()</p>	<pre>public class Password { private String value; private int access; public Password(String s){ value = s; } public String get(){ access++; return value; } } public class CheckedPassword extends Password { public CheckedPassword(String s){ super(s); } public boolean strong(){ String t = get(); return t.length() > 3 && !t.equals(t.toUpperCase()) && !t.equals(t.toLowerCase()); } }</pre>
<p>Assume <*1> is filled in correctly.</p>	
<p>QUESTION 10</p> <p>What is output by the code to the right?</p> <p>A. true1212</p> <p>B. false1212</p> <p>C. true1213</p> <p>D. true12121</p> <p>E. true1</p>	<pre>// client code Password p; p = new CheckedPassword("1212"); String result = <*1> + p.get(); System.out.print(result);</pre>
<p>QUESTION 11</p> <p>What is output by the code to the right?</p> <p>A. 31 B. 28 C. 29</p> <p>D. 19 E. 0</p>	<pre>int m = 29; int n = 14; int o = 19; System.out.print(m & n o);</pre>
<p>QUESTION 12</p> <p>What is output by the code to the right?</p> <p>A. -3.0i B. -4.0i C. -4.0</p> <p>D. -3.0 E. 0</p>	<pre>System.out.print(Math.floor(-Math.sqrt(10)));</pre>

<p>QUESTION 13</p> <p>What is output by the code to the right?</p> <p>A. Tech UT A&M B. Tech UTA&M</p> <p>C. TechUT A&M D. TechUT</p> <p>E. TechUTA&M</p>	<pre>System.out.print("Tech"); System.out.println("UT"); System.out.print("A&M");</pre>
<p>QUESTION 14</p> <p>What is output by the code to the right? Note: each ' ' character below represents a single space in the actual output.</p> <p>A. names: bill ted</p> <p>B. names: ted bill</p> <p>C. names: ted bill</p> <p>D. names: ted bill</p> <p>E. bill tednames:</p>	<pre>String n1 = "ted"; String n2 = "bill"; int mx = Math.max(n1.length(), n2.length()) + 1; String ex = "%1\$" + mx + "s"; String form = "names:" + ex; form += "%2\$" + mx + "s"; System.out.printf(form, n1, n2);</pre>
<p>QUESTION 15</p> <p>What is returned by the method call tough(6)?</p> <p>A. 2 B. 8 C. 78</p> <p>D. 93 E. 243</p>	<pre>public int tough(int x){ if(x <= 2) return 3; else return 2 + tough(x - 1) + tough(x - 1); }</pre>
<p>QUESTION 16</p> <p>What is output by the code to the right?</p> <p>A. 1 B. 12 C. 55</p> <p>D. 66 E. 78</p>	<pre>String stars = ""; for(int k = 1; k <= 12; k++) for(int j = k; j <= 12; j++) stars += "*"; System.out.print(stars.length());</pre>
<p>QUESTION 17</p> <p>What is output by the code to the right?</p> <p>A. 3067 B. 3613 C. 3667</p> <p>D. 4256 E. 4267</p>	<pre>int x2 = 5; int y2 = 6; String r = (++x2 * y2++) + "" + x2 + y2; System.out.print(r);</pre>
<p>QUESTION 18</p> <p>What is output by the code to the right?</p> <p>A. 0 B. 3 C. 5</p> <p>D. 15 E. 21</p>	<pre>int[] hts = {5, 1, 5, 7, 0, 3}; int total = 0; for(int i : hts) total += i; System.out.print(total);</pre>

<p>QUESTION 19</p> <p>Which of the following can replace <code><*1></code> in the code to the right so that the code segment compiles without error?</p> <p>I. Object II. List III. Collection</p> <p>A. I only B. II only C. III only</p> <p>D. I, II, and III E. None of I, II, or III</p>	<pre>List<String> sample; sample = new <*1><String>();</pre>
<p>QUESTION 20</p> <p>What is returned by the method call <code>tellMe(new int[]{7, 1, 3, 2, 10, 3})</code>?</p> <p>A. -9 B. 1 C. 9</p> <p>D. 11 E. 25</p>	<pre>// pre: data != null, data.length % 2 == 0 // post: question 21 public int tellMe(int[] data) { int x = data[0]; int y = data[0]; for(int i = 0; i < data.length; i += 2) { int z1 = data[i]; int z2 = data[i + 1]; if(z1 > z2) { int temp = z1; z1 = z2; z2 = temp; } x = (z1 < x) ? z1 : x; y = (z2 > y) ? z2 : y; } return y - x; }</pre>
<p>QUESTION 21</p> <p>Which of the following best describes the post condition for method <code>tellMe</code>?</p> <p>A. returns the difference of the maximum and minimum elements in <code>data</code></p> <p>B. returns the sum of the maximum and minimum elements in <code>data</code>.</p> <p>C. returns the minimum value in <code>data</code>.</p> <p>D. returns the maximum value in <code>data</code>.</p> <p>E. returns the sum of all the elements in <code>data</code></p>	<pre>int[] sm = {7, 1, 3, 2, 10, 3}; ArrayList<<*1>> hold; hold = new ArrayList<<*1>>(); for(int i : sm) hold.add(i); Iterator<<*1>> it = hold.iterator(); it.next(); it.next(); it.remove(); it.next(); it.remove(); System.out.print((int) it.next());</pre>
<p>QUESTION 22</p> <p>Which of the following can replace <code><*1></code> in the code to the right so that the code segment compiles without error?</p> <p>A. double B. Double C. int</p> <p>D. Integer E. String</p>	<pre>int[] sm = {7, 1, 3, 2, 10, 3}; ArrayList<<*1>> hold; hold = new ArrayList<<*1>>(); for(int i : sm) hold.add(i); Iterator<<*1>> it = hold.iterator(); it.next(); it.next(); it.remove(); it.next(); it.remove(); System.out.print((int) it.next());</pre>
<p>Assume <code><*1></code> is filled in correctly.</p>	
<p>QUESTION 23</p> <p>What is output by the code to the right?</p> <p>A. 1 B. 2 C. 3</p> <p>D. 7 E. 10</p>	<pre>int[] sm = {7, 1, 3, 2, 10, 3}; ArrayList<<*1>> hold; hold = new ArrayList<<*1>>(); for(int i : sm) hold.add(i); Iterator<<*1>> it = hold.iterator(); it.next(); it.next(); it.remove(); it.next(); it.remove(); System.out.print((int) it.next());</pre>

<p>QUESTION 24</p> <p>What is returned by the method call <code>rec2(13)</code>?</p> <p>A. 30 B. 10 C. 7</p> <p>D. There is no output due to a syntax error in method <code>rec2</code>.</p> <p>E. There is no output due to a runtime error.</p>	<pre>public int rec2(int n) { if(n >= 20) return n / 2; else return n / 3 + rec2(n + 2); }</pre>																		
<p>QUESTION 25</p> <p>Which sorting algorithm does method <code>sort</code>, shown to the right, implement?</p> <p>A. heap sort B. insertion sort C. selection sort</p> <p>D. radix sort E. merge sort</p>	<pre>public void sort(List<Double> data) { int lim = data.size(); for(int i = 0; i < lim; i++) { int m = i; for(int j = i + 1; j < lim; j++) if(data.get(j) < data.get(m)) m = j; data.set(i, data.get(m)); } }</pre>																		
<p>QUESTION 26</p> <p>What is the order (Big O) of method <code>sort</code> shown to the right, given the following kinds of Lists? Pick the most restrictive correct set of answers.</p> <table><thead><tr><th></th><th>LinkedList</th><th>ArrayList</th></tr></thead><tbody><tr><td>A.</td><td>$O(N^3)$</td><td>$O(N^3)$</td></tr><tr><td>B.</td><td>$O(N^2)$</td><td>$O(N^2)$</td></tr><tr><td>C.</td><td>$O(N \log N)$</td><td>$O(N^2)$</td></tr><tr><td>D.</td><td>$O(N^3)$</td><td>$O(N^2)$</td></tr><tr><td>E.</td><td>$O(N!)$</td><td>$O(N!)$</td></tr></tbody></table>		LinkedList	ArrayList	A.	$O(N^3)$	$O(N^3)$	B.	$O(N^2)$	$O(N^2)$	C.	$O(N \log N)$	$O(N^2)$	D.	$O(N^3)$	$O(N^2)$	E.	$O(N!)$	$O(N!)$	
	LinkedList	ArrayList																	
A.	$O(N^3)$	$O(N^3)$																	
B.	$O(N^2)$	$O(N^2)$																	
C.	$O(N \log N)$	$O(N^2)$																	
D.	$O(N^3)$	$O(N^2)$																	
E.	$O(N!)$	$O(N!)$																	
<p>QUESTION 27</p> <p>What replaces <code><*1></code> in the code the right so that the output is <code>[0, 7, 9]</code>?</p> <p>A. <code>Collections.sort(nums)</code></p> <p>B. <code>Arrays.sort(nums)</code></p> <p>C. <code>nums.sort()</code></p> <p>D. <code>ArrayList.sort(nums)</code></p> <p>E. More than one of A through D is correct.</p>	<pre>ArrayList<Integer> nums; nums = new ArrayList<Integer>(); nums.add(9); nums.add(0); nums.add(7); <*1>; System.out.print(nums);</pre>																		
<p>QUESTION 28</p> <p>Which of the following is not a Java keyword?</p> <p>A. finally B. throw C. throws D. foreach E. Two or more of the choices in answers A through D are not keywords.</p>																			
<p>QUESTION 29</p> <p>What is output by the code to the right?</p> <p>A. -712131 B. -7312</p> <p>C. 123-7 D. 13112-7</p> <p>E. There is no output due to a syntax error.</p>	<pre>PriorityQueue<Integer> pq; pq = new PriorityQueue<Integer>(); pq.add(12); pq.add(-7); pq.add("131".length()); while(!pq.isEmpty()) System.out.print(pq.remove());</pre>																		

QUESTION 30

Which of the following expressions can replace `<*1>` in method `b` to the right so that the expression gives the result of `x` divided by 2 using integer division?

- I. `x / 2`
- II. `x >> 2`
- III. `x % 2`

- A. I only
- B. II only
- C. III only
- D. I and II only
- E. II and III only

```
public int a(int x) {
    System.out.print(x * 2);
    return x * 2;
}

public int b(int x) {
    System.out.print(<*1>);
    return <*1>;
}
```

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

QUESTION 31

What is output by the code to the right?

- A. 12122false
- B. false122122
- C. false12122
- D. 122122false
- E. 12false8true122false

```
// client code
int y4 = 4;
int z4 = 6;
System.out.print( y4 >= a(z4)
    && z4 <= b(y4) || a(z4) == b(y4));
```

QUESTION 32

What is returned by the method call `trace2(17, 42)`?

- A. 26
- B. 16
- C. 8
- D. There is no output due to a syntax error.
- E. There is no output due to a runtime error.

```
// pre: 0 <= s <= e
public int trace2(int s, int e) {
    int t = 0;
    for(int i = s; i <= e; i++)
        if(helper(i))
            t++;
    return t;
}

// pre: x > 0
public boolean helper(int x) {
    String s = x + "";
    for(int i = 1; i < s.length(); i++)
        if( s.charAt(i) >= s.charAt(i - 1) )
            return false;
    return true;
}
```

QUESTION 33

What is output by the code to the right?

- A. [1]
- B. [1, 1]
- C. [5]
- D. [1, 1, 5]
- E. There is no output due to a syntax error.

```
Set<Integer> s1 = new TreeSet<Integer>();
Set<Integer> s2 = new TreeSet<Integer>();
s1.add(1);
s1.add(1);
s2.add(1);
s2.add(5);

s2.retainAll(s1);
System.out.print(s2);
```

QUESTION 34

What is returned by the method call `srch(m, 2, 2)` if `m` is the matrix shown below?

11	15	3	12	11	16
11	10	4	7	9	14
12	3	6	5	3	1
1	15	13	13	14	1

- A. 3 B. 1 C. 0
D. 4 E. 2

QUESTION 35

What is returned by the method call `srch(m, 2, 4)` if `m` is the matrix shown below?

10	15	17	21	23	20	20	22
17	10	16	18	20	20	18	31
7	12	15	17	27	24	25	30
19	17	16	18	26	10	5	1
-7	13	17	10	33	27	27	30

- A. 9 B. 6 C. 3
D. 2 E. 1

QUESTION 36

What is the order (Big O) of method `fillCol` to the right, given the following kinds of Collections? Pick the most restrictive correct set of answers.

LinkedList TreeSet

- | | |
|------------------|---------------|
| A. $O(N)$ | $O(N)$ |
| B. $O(N)$ | $O(N \log N)$ |
| C. $O(N)$ | $O(N^2)$ |
| D. $O(N \log N)$ | $O(N^2)$ |
| E. $O(N^2)$ | $O(N^2)$ |

```
public int srch(int[][] m, int r, int c) {
    if(r == 0 || c == 0 || r == m.length - 1
        || c == m[0].length - 1)
        return 1;
    else {
        int[][] ds = {{-1, 0, 1, 0, -1},
                      {0, 1, 0, -1, 1}};

        int t = 0;
        for(int i = 0; i < ds[0].length; i++) {
            int r1 = r + ds[0][i];
            int c1 = c + ds[1][i];
            if( m[r1][c1] < m[r][c])
                t += srch(m, r1, c1);
        }
        return t;
    }
}
```

```
// pre: col.isEmpty()
public void fillCol(int n,
                   Collection<Integer> col) {

    for(int i = 0; i < n; i++)
        col.add(i);
}
```

QUESTION 37

A class that represents a graph uses an adjacency matrix to represent the connections between vertices. What are the space requirements for a graph that contains N nodes with an average of M connections per node? Pick the most restrictive correct answer.

- A. $O(N^2)$ B. $O(NM)$ C. $O(N^2M)$ D. $O(N)$ E. $O(N^2 + \log M)$

QUESTION 38

Which of the following can replace `<*1>` in the code to the right so that `dir` stores a negative integer if `obj` is less than the element at `d[p]`, 0 if the two objects are equal, and a positive integer if `obj` is greater than the element at `d[p]`?

- I. `obj.equals(d[p])`
- II. `d[p].compareTo(obj)`
- III. `obj.compareTo(d[p])`

- A. I only
- B. II only
- C. III only
- D. I and II only
- E. I and III only

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

QUESTION 39

What is output by the following client code?

```
Structure<Integer> s1;
s1 = new Structure<Integer>();
int[] vals = {14, 41, -5, 14, 7, -5};
for(int ins : vals)
    s1.add(ins);
s1.show();
```

- A. -5 7 14 41
- B. 41 -5 7 14 -5 14
- C. 7 -5 41 14
- D. -5 -5 7 14 14 41
- E. 41 7 -5 14

QUESTION 40

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

- A. an array based list
- B. a stack
- C. a linked list
- D. a max heap
- E. a binary search tree

```
public class Structure
    <E extends Comparable<? super E>> {

    private E[] d;

    public Structure() { d = getArray(1); }

    public void show() {show(0);}

    private void show(int p) {
        if(p < d.length && d[p] != null) {
            show(r(p));
            show(l(p));
            System.out.print(d[p] + " ");
        }
    }

    public void add(E obj) {
        int p = getPos(obj);
        if(p >= d.length)
            resize();
        d[p] = obj;
    }

    public boolean present(E obj) {
        int p = getPos(obj);
        return p < d.length && d[p] != null;
    }

    private int getPos(E obj) {
        int p = 0;
        while(p < d.length && d[p] != null) {
            int dir = <*1>;
            if(dir < 0)
                p = l(p);
            else if(dir > 0)
                p = r(p);
            else
                return p;
        }
        return p;
    }

    private E[] getArray(int size) {
        return (E[]) new Comparable[size];
    }

    private void resize() {
        E[] temp = getArray(d.length * 2 + 1);
        for(int i = 0; i < d.length; i++)
            temp[i] = d[i];
        d = temp;
    }

    private int l(int i) { return 2 * i + 1;}
    private int r(int i) { return 2 * i + 2;}
}
```

Computer Science Answer Key

UIL State 2011

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

Notes: The clause "Choose the most restrictive correct answer." is necessary because per the formal definition of Big O, an algorithm that is $O(N^2)$ is also $O(N^3)$, $O(N^4)$, and so forth.

9. The declared type of the variable `p` is `Password`. When a method call is made on a variable that is not in the declared type or its ancestors, it must be downcast to the proper type.

22. Although `ints` are autocast to `doubles`, `ints` are not autoboxed to `Doubles`.

26. Although selection sort is usually $O(N^2)$, the `get` method for `LinkedLists` is $O(N)$. With N^2 calls to `get` the sort method shown is $O(N^3)$ for `LinkedLists`.

34 and 35. Method `srch` returns the number of paths from the cell (assuming it is not on the border of the matrix) to the border of the matrix. Movement is only allowed to cells with a smaller value than the current cell. For question 34 the two paths contains these cells: [6, 4, 3], [6, 5, 3, 1]. For question 35 the 9 paths are: [27, 20, 18, 17, 15, 12, 7]
[27, 20, 18, 16, 15, 12, 7], [27, 24, 10, 5, 1], [27, 26, 10, 5, 1], [27, 26, 18, 17, 15, 12, 7], [27, 26, 18, 10],
[27, 26, 18, 16, 15, 12, 7], [27, 26, 24, 10, 5, 1], and [27, 17, 15, 12, 7]

36. The Java `TreeSet` uses a balanced binary search tree. This ensure a worst case performance of $O(N \log N)$ when adding N values, even if the values are added in sorted order. (This performance is $O(N^2)$ for a traditional binary search tree.)