

# A+ Computer Science

## M/C Written Test

### General Directions:

- 1) DO NOT OPEN EXAM UNTIL TOLD TO DO SO.
- 2) **NO CALCULATORS of any kind may be used.**
- 3) 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 forty-five 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. You may 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 except on the answer sheet or Scantron card which is 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 (i.e. `error` is an answer choice). Ignore any typographical errors and assume any undefined variables are defined as used.**
- 9) A reference to commonly used Java classes is provided with the test and you may use this reference 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 12 Packages and classes (e.g. `.lang`, `.util`, `System`, `Math`, `Double`, etc.) are included in any programs or code segments that refer to methods from these classes and/or 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 each incorrect answer.

**For more Computer Science practice tests and materials,**

**go to [www.apluscompsci.com](http://www.apluscompsci.com)**

**Standard Classes and Interfaces — Supplemental Reference****class java.lang.Object**

- o boolean equals(Object other)
- o String toString()
- o int hashCode()

**interface java.lang.Comparable<T>**

- o int compareTo(T other)  
Return value < 0 if this is less than other.  
Return value = 0 if this is equal to other.  
Return value > 0 if this is greater than other.

**class java.lang.Integer implements****Comparable<Integer>**

- o Integer(int value)
- o int intValue()
- o boolean equals(Object obj)
- o String toString()
- o int compareTo(Integer anotherInteger)
- o static int parseInt(String s)

**class java.lang.Double implements****Comparable<Double>**

- o Double(double value)
- o double doubleValue()
- o boolean equals(Object obj)
- o String toString()
- o int compareTo(Double anotherDouble)
- o static double parseDouble(String s)

**class java.lang.String implements****Comparable<String>**

- o int compareTo(String anotherString)
- o boolean equals(Object obj)
- o int length()
- o String substring(int begin, int end)  
Returns the substring starting at index begin and ending at index (end - 1).
- o String substring(int begin)  
Returns substring(from, length()).
- o int indexOf(String str)  
Returns the index within this string of the first occurrence of str. Returns -1 if str is not found.
- o int indexOf(String str, int fromIndex)  
Returns the index within this string of the first occurrence of str, starting the search at the specified index.. Returns -1 if str is not found.
- o charAt(int index)
- o int indexOf(int ch)
- o int indexOf(int ch, int fromIndex)
- o String toLowerCase()
- o String toUpperCase()
- o String[] split(String regex)
- o boolean matches(String regex)

**class java.lang.Character**

- o static boolean isDigit(char ch)
- o static boolean isLetter(char ch)
- o static boolean isLetterOrDigit(char ch)
- o static boolean isLowerCase(char ch)
- o static boolean isUpperCase(char ch)
- o static char toUpperCase(char ch)
- o static char toLowerCase(char ch)

**class java.lang.Math**

- o static int abs(int a)
- o static double abs(double a)
- o static double pow(double base, double exponent)
- o static double sqrt(double a)
- o static double ceil(double a)
- o static double floor(double a)
- o static double min(double a, double b)
- o static double max(double a, double b)
- o static int min(int a, int b)
- o static int max(int a, int b)
- o static long round(double a)
- o static double random()  
Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.

**interface java.util.List<E>**

- o boolean add(E e)
- o int size()
- o Iterator<E> iterator()
- o ListIterator<E> listIterator()
- o E get(int index)
- o E set(int index, E e)  
Replaces the element at index with the object e.
- o void add(int index, E e)  
Inserts the object e at position index, sliding elements at position index and higher to the right (adds 1 to their indices) and adjusts size.
- o E remove(int index)  
Removes element from position index, sliding elements at position (index + 1) and higher to the left (subtracts 1 from their indices) and adjusts size.

**class java.util.ArrayList<E> implements List<E>****class java.util.LinkedList<E> implements****List<E>, Queue<E>**

Methods in addition to the List methods:

- o void addFirst(E e)
- o void addLast(E e)
- o E getFirst()
- o E getLast()
- o E removeFirst()
- o E removeLast()

**class java.util.Stack<E>**

- o boolean isEmpty()
- o E peek()
- o E pop()
- o E push(E item)

**interface java.util.Queue<E>**

- o boolean add(E e)
- o boolean isEmpty()
- o E peek()
- o E remove()

**class java.util.PriorityQueue<E>**

- o boolean add(E e)
- o boolean isEmpty()
- o E peek()
- o E remove()

**interface java.util.Set<E>**

- o boolean add(E e)
- o boolean contains(Object obj)
- o boolean remove(Object obj)
- o int size()
- o Iterator<E> iterator()
- o boolean addAll(Collection<? extends E> c)
- o boolean removeAll(Collection<?> c)
- o boolean retainAll(Collection<?> c)

**class java.util.HashSet<E> implements Set<E>**

**class java.util.TreeSet<E> implements Set<E>**

**interface java.util.Map<K,V>**

- o Object put(K key, V value)
- o V get(Object key)
- o boolean containsKey(Object key)
- o int size()
- o Set<K> keySet()
- o Set<Map.Entry<K, V>> entrySet()

**class java.util.HashMap<K,V> implements Map<K,V>**

**class java.util.TreeMap<K,V> implements Map<K,V>**

**interface java.util.Map.Entry<K,V>**

- o K getKey()
- o V getValue()
- o V setValue(V value)

**interface java.util.Iterator<E>**

- o boolean hasNext()
- o E next()
- o void remove()

**interface java.util.ListIterator<E> extends  
java.util.Iterator<E>**

Methods in addition to the Iterator methods:

- o void add(E e)
- o void set(E e)

**class java.lang.Exception**

- o Exception()
- o Exception(String message)

**class java.util.Scanner**

- o Scanner(InputStream source)
- o boolean hasNext()
- o boolean hasNextInt()
- o boolean hasNextDouble()
- o String next()
- o int nextInt()
- o double nextDouble()
- o String nextLine()
- o Scanner useDelimiter(String pattern)

Note: Correct responses are based on **Java SE Development Kit 20 (JDK 20)** 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 20 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.*`**

<b>QUESTION 1</b>	
What is $4C_{16} - 101011_2$ ?	
A. 201 <sub>4</sub> B. 10011 <sub>2</sub> C. 31 <sub>10</sub> D. 43 <sub>8</sub> E. 19 <sub>16</sub>	
<b>QUESTION 2</b>	
What is output by the code to the right?	<code>out.println(45 + 7 * 4 - 3 / 2);</code>
A. 102      B. 102.5      C. 72      D. 71.5      E. 26	
<b>QUESTION 3</b>	
What is output by the code to the right?	
A. 76.3212 B. 76.3212%.2f C. %.3f76.3212 D. 76.32 E. 76	<code>out.printf("%.2f", 76.3212);</code>
<b>QUESTION 4</b>	
What is output by the code to the right?	
A. chocolate peanut butter B. chocolatepeanut butter C. chocolatepeanut D. chocolate peanut E. chocolatepeanutbutter	<code>String s = "chocolate"; String b = "peanut butter"; out.println(s.concat(b.substring(0, 6)));</code>
<b>QUESTION 5</b>	
What is output by the code to the right?	<code>out.println(true&amp;&amp;false  true&amp;&amp;false&amp;&amp;true);</code>
A. true      B. false	
<b>QUESTION 6</b>	
What is output by the code to the right?	
A. 12 B. 12.0 C. 12.3 D. 12.4 E. 13	<code>out.print(Math.floor(12.36));</code>
<b>QUESTION 7</b>	
What is output by the code to the right?	<code>String luigi = "Luigi"; int mario = 5; double peach = 10.5; char toad = 'a'; out.println(mario + toad + peach + luigi);</code>
A. 5a10.5Luigi              B. 10210.5Luigi C. 112.5Luigi D. Output cannot be determined. E. There is no output due to a compile error.	

<p><b>QUESTION 8</b></p> <p>What is output by the code to the right?</p> <p>A. 0            B. 1            C. 2            D. 3</p> <p>E. 1230</p>	<pre>var a = 8; var test = switch(a) { case 8 -&gt; 1; case 9 -&gt; 2; case 10 -&gt; 3; default -&gt; 0; }; out.println(test);</pre>
<p><b>QUESTION 9</b></p> <p>How many asterisks are output by the code to the right?</p> <p>A. 5            B. 4            C. 15            D. 10            E. 0</p>	<pre>for(int i = 0; i &lt; 5; i++) {     out.println("*".repeat(i)); }</pre>
<p><b>QUESTION 10</b></p> <p>What is the output by the code to the right?</p> <p>A. [1, 2, 3, 4, 5]   B. [1, 2, 0, 4, 5]</p> <p>C. [1, 2, 3, 0, 5]   D. [0, 2, 3, 4, 5]</p> <p>E. Output cannot be determined until runtime.</p>	<pre>int [] gerald = {1, 2, 3, 4, 5}; int [] herald = gerald; herald[3] = 0; out.println(gerald);</pre>
<p><b>QUESTION 11</b></p> <p>What is the output by the code to the right?</p> <p>A. beebobeebop</p> <p>B. beebobeebop\nbeebobeebeebop</p> <p>C. beebobeebeebop</p> <p>D. There is no output due to a syntax error.</p> <p>E. There is no output due to a runtime exception.</p>	<pre>Scanner file = new Scanner("beebobeebop\nbeebobeebeebop "); file.nextLine(); String s = file.next(); out.println(s);</pre>
<p><b>QUESTION 12</b></p> <p>What is the output by the code to the right?</p> <p>A. 24            B. 4            C. 120            D. 3</p> <p>E. There is no output due to a runtime exception.</p>	<pre>int i = 0; int b = 1; while(i &lt; 4) {     i++;     b += i*b; } out.println(b);</pre>
<p><b>QUESTION 13</b></p> <p>What is order of precedence for the operations on the right from highest precedence to lowest precedence?</p> <p>A. I, II, III</p> <p>B. I, III, II</p> <p>C. II, III, I</p> <p>D. III, I, II</p> <p>E. III, II, I</p>	<p>I. &amp;(bitwise)</p> <p>II.  (bitwise)</p> <p>III. ^(bitwise)</p>
<p><b>QUESTION 14</b></p> <p>What is the output by the code to the right?</p> <p>A. 8            B. 64            C. 32            D. 16            E. 1</p>	<pre>out.println(Byte.BYTES);</pre>

<p><b>QUESTION 15</b></p> <p>What is the output by the code to the right?</p> <p>A. true</p> <p>B. false</p> <p>C. 0</p> <p>D. There is no output due to a compile error.</p> <p>E. There is no output due to a runtime exception.</p>	<pre>ArrayList &lt;String&gt; ray; ray = new ArrayList&lt;String&gt;(); ray.add("practice.apluscompsci.com"); ray.add("The BEST UIL Prep!"); ray.add("onion"); ray.add("carrot"); ray.add("celery"); ray.add("garlic"); out.println(ray.contains("celery"));</pre>
<p><b>QUESTION 16</b></p> <p>What is the output by the code to the right?</p> <p>A. 22      B. 24      C. 6      D. 12      E. 18</p>	<pre>out.println(6 &amp; 12   18);</pre>
<p><b>QUESTION 17</b></p> <p>What is the output by the code to the right?</p> <p>A. true      B. false</p>	<pre>boolean a = true; boolean b = true; out.print(a&amp;&amp;b&amp;&amp;!true              !a&amp;&amp;!b  true&amp;&amp;(a&amp;&amp;b));</pre>
<p><b>QUESTION 18</b></p> <p>What is the output by the code to the right?</p> <p>A. 8</p> <p>B. 80.2</p> <p>C. cheese</p> <p>D. There is no output due to a compile error.</p> <p>E. There is no output due to a runtime exception.</p>	<pre>ArrayList &lt;Object&gt; ar; ar = new ArrayList &lt;Object&gt;(); ar.add(8); ar.add(80.2); ar.add("cheese"); Object[] a = ar.toArray(); out.print(a[0]);</pre>
<p><b>QUESTION 19</b></p> <p>What is the output by the code to the right?</p> <p>A. 86      B. 8      C. 88      D. 3</p> <p>E. There is no output due to a runtime exception.</p>	<pre>int a = 83; int b = 5; out.print(a % b + b % a);</pre>
<p><b>QUESTION 20</b></p> <p>How many asterisks are printed by the code to the right?</p> <p>A. 6      B. 14      C. 62      D. 126      E. 30</p>	<pre>int a = 0; int b = 1; do {     b*=2;     a++;     for(int i = 0; i &lt; b; i++) {         out.print("*");     } }while(a&lt;4);</pre>

<p><b>QUESTION 21</b></p> <p>What is the output of the function call <code>jelly(2, 2)</code>?</p> <p>A. 1      B. 2      C. 3      D. 4      E. 0</p>	<p><b>// Use the code below to answer // questions 21-22.</b></p> <pre>public static int jelly(int a, int b) {     if(a &gt; b &amp;&amp; a % 2 == 0) return 1;      if(a &gt; b) return 2;      if(a &lt; b &amp;&amp; b % 2 == 0) return 3;      if(a &lt; b) return 4;      return 0; }</pre>
<p><b>QUESTION 22</b></p> <p>What is the output of the function call <code>jelly(3, 5)</code>?</p> <p>A. 1      B. 2      C. 3      D. 4      E. 0</p>	
<p><b>QUESTION 23</b></p> <p>What should replace <b>&lt;code 1&gt;</b> in the code to the right in order for <code>lizard</code> to be a subclass of <code>reptile</code>?</p> <p>A. defines B. extends C. implements D. inherits E. Nothing, this space can be left blank.</p>	<p><b>// Use the code below to answer // questions 23-26.</b></p> <pre>class reptile{     public String species;     public int legs;      public reptile() {         legs = 0;         species = "long";     }     public reptile(int l, String s) {         legs = l;         species = s;     }     public String run() {         return switch(legs) {             case 0 -&gt; "no";             default -&gt; "yes";         };     } }  class lizard <b>&lt;code 1&gt;</b> reptile{     public int tails;      public lizard(int t) {         <b>&lt;code 2&gt;</b>         tails = t;     }      public String run() {         return "very fast";     } }</pre>
<p><b>QUESTION 24</b></p> <p>What should replace <b>&lt;code 2&gt;</b> in the code to the right in order for <code>lizard</code> constructor to function properly?</p> <p>A. <code>super(4, "lizard");</code> B. <code>super();</code> C. <b>&lt;code 2&gt;</b> can be left blank. D. Only A and B. E. All of A, B, and C.</p>	

**QUESTION 25**

Assuming the blanks above have been properly filled, what is the output of the code marked **//line 1**?

- A. yes
- B. no
- C. very fast
- D. There is no output due to a compile error.
- E. There is no output due to a runtime exception.

```
// use the code above and the code
// below to answer questions 25-26
```

```
// client code
```

```
reptile bob = new reptile();
reptile john = new lizard(1);
out.println(bob.run()); // line 1
out.println(john.run()); // line 2
```

**QUESTION 26**

Assuming the blanks above have been properly filled and any errors fixed, what is the output of the code marked **//line 2**?

- A. yes
- B. no
- C. very fast
- D. There is no output due to a compile error.
- E. There is no output due to a runtime exception.

**QUESTION 27**

What is the result of the function call `recu(2,2)`?

- A. 13      B. 4      C. 7
- D. 10      E. 12

```
public static int recu(int a, int b) {
    if(a > 0 && b > 0) {
        int temp = 0;
        if(a/2 > b)
            temp += a/2 + recu(a - 1, b);
        if(a/2 < b)
            temp += a + recu(a - 1, b);
        if(b/2 > a)
            temp += b/2 + recu(a - 1, b - 1);
        if(b/2 < a)
            temp += b + recu(a, b - 1);
        return temp;
    }
    return a+b;
}
```

**QUESTION 28**

What is the result of the function call `recu(2,3)`?

- A. 22
- B. 17
- C. 19
- D. 63
- E. There is no output due to a runtime exception.

**QUESTION 29**

What is output by the code to the right?

- A. 234      B. 94      C. 136
- D. 1011110      E. 5e

```
out.print(Integer.toHexString(94));
```



**QUESTION 30**

What could replace **<code 1>** so that the map is instantiated without causing a compilation error?

- A. Map<String, int>()
- B. TreeMap<String, int>()
- C. TreeMap()
- D. Both A and B.
- E. Both B and C.

```
Map <String, Integer> jon;
jon = new <code 1> ;
```

**QUESTION 31**

What is output by the code to the right?

- A. [1, 1, 1, 1]
- B. [1, 1, 1, 0]
- C. [1, 1, 0, 0]
- D. [0, 0, 0, 0]
- E. There is no output due to a compile error.

```
int[][] jorge = new int[4][4];
for(int i = 0; i < 4; i++) {
    for(int j = 0; j <=i; j++) {
        jorge[i][j] = 1;
    }
}
out.print(Arrays.toString(jorge[2]));
```

**QUESTION 32**

What is output by the code to the right?

- A. 9
- B. 3
- C. 4
- D. 5
- E. 0

```
Stack <Integer> stack;
stack = new Stack <Integer>();
stack.push(9); stack.push(3);
stack.peek();
stack.push(4); stack.add(5);
stack.pop();
out.print(stack.peek());
```

**QUESTION 33**

What is output by the code to the right?

- A. 45
- B. 32
- C. 12
- D. 43
- E. 34

```
LinkedList <Integer> ll = new
LinkedList<Integer>();
ll.add(45);ll.add(32);ll.add(78);
ll.add(34); ll.add(43); ll.add(12);
ll.poll();
out.print(ll.poll());
```

**QUESTION 34**

What is output by the code to the right?

- A. true    B. false

```
String s = "over there";
out.print(s.matches("\\D+\\s+\\w+"));
```

**QUESTION 35**

What is output by the code to the right?

- A. 7    B. 16    C. 23    D. 11  
E. There is no output due to a compile error.

```
out.print(3^4|19&16<<1);
```

**QUESTION 36**

What is the average time complexity for insertion on a singly-linked list of size n?

- A.  $O(1)$     B.  $O(n)$     C.  $O(\log(n))$     D.  $O(n \log(n))$     E.  $O(n^2)$

**QUESTION 37**

Which of the following Java boolean expressions is equivalent to the truth table on the right?

- A.  $A \ || \ B \ \wedge \ A$   
B.  $B \ || \ B \ \wedge \ A$   
C.  $!A \ || \ B \ \wedge \ A$   
D.  $B \ || \ !A \ \wedge \ B$   
E.  $A \ || \ A \ \wedge \ !B$

A	B	Result
True	True	True
True	False	False
False	True	True
False	False	True

**QUESTION 38**

How many edges are in a complete undirected graph with 5 nodes?

- A. 20  
B. 15  
C. 10  
D. 5  
E. 9

**QUESTION 39**

What is the value of the prefix expression to the right? Assume decimal division.

```
- + 4 * 6 2 / 8 2
```

**QUESTION 40**

After inserting the following values in the following order, how many leaves are in the binary search tree?

13 45 27 12 43 19 23 26