

UIL COMPUTER SCIENCE WRITTEN TEST

2024 STATE

MAY 2024

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.

This page intentionally left blank

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 – 2024 STATE

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.*;`**

Question 1

Which of the following has the most 1's in its binary representation?

- A) 7654_8 B) AAA_{16} C) 32132_4 D) 444444_8 E) 63_{10}

Question 2

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

- A) 6 B) 8 C) 10 D) 11 E) 12

```
out.print(22 % 30 * 4 / 10);
```

Question 3

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

- A) MNO B) MN C) NOP D) NO E) LMNOP

```
String Alpha = "ABCDEFGHJKLMN";  
String Beta = "NOPQRSTUVWXYZ";  
String Alphabet = Alpha+Beta;  
String Z = Alphabet.substring(10,15);  
out.print(Z.substring(3));
```

Question 4

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

- A) true
B) false

```
boolean A = true;  
boolean B = !A || A;  
boolean C = !B && A;  
out.print(B ^ C);
```

Question 5

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

- A) 24 B) 28 C) 31 D) 50 E) 64

```
int T = 10 + 20 * 30 / 40;  
int R = 40 + 30 * 20 / 10;  
int H = T + R >> 2;  
out.print(H);
```

Question 6

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

- A) 18 B) 19 C) 20 D) 21 E) 22

```
double A = Math.sqrt(10);  
double B = Math.sqrt(5);  
double C = Math.sqrt(2);  
int D = (int)(A*B*C*C*C);  
out.print(D);
```

Question 7

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

- A) 12 10 11
B) 9 9 9
C) 10 11 8
D) 10 10 10
E) 10 9 10

```
int A = 11; int B = 10; int C = 12;  
if(A>C) A++; B--;  
if(A>B) B++; C--;  
if(B>C) C++; A--;  
if(C>B) A++; B--;  
if(B>A) B++; C--;  
if(C>A) C++; A--;  
out.print(A+" "+B+" "+C);
```

<p>Question 8</p> <p>What is the output of the code segment to the right?</p> <p>A) 12 B) 14 C) 18 D) 23 E) 24</p>	<pre>int[] Numbers = {8,6,7,5,3,0,9}; int C = Numbers[1]; int D = C; for(int x=Numbers[4]; x<D; x++) C+=Numbers[x]; out.print(C);</pre>
<p>Question 9</p> <p>What is the output of the code segment to the right?</p> <p>A) D B) DEF C) DEFG D) EFG E) DEFGHIJ</p>	<pre>String St = "ABCDEFGHJIJ"; for(int x=1; x<=3; x++) { int T = St.length(); St = St.substring(1,T-1); } out.print(St);</pre>
<p>Question 10</p> <p>What is output by the code segment to the right?</p> <p>A) 26 B) -26 C) 12 D) -12 E) -32</p>	<pre>out.print((int)'A' - (int)'a');</pre>
<p>Question 11</p> <p>What is the output of the code segment to the right?</p> <p>A) SO VERY TIRED B) SO TIRED C) VERY TIRED D) VERY HUNGRY E) SO HUNGRY</p>	<pre>int A = 4; int B = 7; int C = 8; out.print((A<B)?"SO ":"VERY "); out.print((C<B)?"TIRED ":"HUNGRY ");</pre>
<p>Question 12</p> <p>What is the output of the code segment to the right?</p> <p>A) 15 B) 28 C) 29 D) 30 E) 31</p>	<pre>int A = 30 & 29; int B = 29 & 28; int C = A B; out.print(C);</pre>
<p>Question 13</p> <p>What is the output of the code segment shown on the right?</p> <p>A) 30 B) 32 C) 35 D) 36 E) 40</p>	<pre>int A = 12345678; int B = A % 10; int C = A / 100; int D = C % 10; int E = C / 100; int F = E % 10; int G = E / 100; out.print(B+D+F+G);</pre>

Question 14

What is output by the code segment to the right?

- A) 32
- B) 36
- C) 40
- D) 75
- E) 90

```
int x = 100;
while (x>75)
    x=x-6;
while (x<90)
    x=x+4;
while (x>40)
    x=x-10;
out.print(x);
```

Question 15

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

- A) 12 B) 13 C) 14 D) 48 E) 54

```
int C = 0;
int[]A = {0,1,10,11,100,101,111};
int[]B = {22,21,20,12,11,10,2,1,0};
for(int x : A)
    for(int y : B)
        if(y/10==x/10)
            C++;
out.print(C);
```

Question 16

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

- A) 1 B) 2 C) 4 D) 6 E) 8

```
int T = 0;
String A = new String("UIL");
String B = new String("UIL");
String C = new String("TEXAS");
if (A==B) T++;
if (B==C) T++;
if (A==C) T++;
if (B==C) T++;
if (A==C) T++;
if (B==C) T++;
if (A.equals(C)) T++;
if (A.equals(B)) T++;
out.print(T);
```

Question 17

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

- A) 20.24
UNIVERSITY
OF TEXAS
INTERSCHOLASTIC
LEAGUE
- B) 20.24
UNIVERSITY OF TEXAS
INTERSCHOLASTIC LEAGUE
- C) 20.23
UNIVERSITY OF TEXAS
INTERSCHOLASTIC LEAGUE
- D) 20.24 UNIVERSITY OF TEXAS
INTERSCHOLASTIC LEAGUE
- E) 20.24
UNIVERSITY
OF TEXAS INTERSCHOLASTIC
LEAGUE

```
double A = 20.237;  
  
out.printf("%.2f ", A);  
out.print("UNIVERSITY ");  
out.println("OF TEXAS ");  
out.print("INTERSCHOLASTIC ");  
out.println("LEAGUE ");
```

Question 18

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

- A) 1 B) 3 C) 5 D) 7 E) 9

```
TreeSet<Integer>House;  
House = new TreeSet<Integer>();  
for(int x=1; x<=20; x++)  
    House.add(x);  
for(int x=15; x<=35; x++)  
    House.add(x);  
for(int x=-12; x<=8; x++)  
    House.add(x);  
int C = 0;  
for(int Bob:House)  
    C = Bob;  
out.print(C%10);
```

Question 19

What is output by the code segment to the right?

- A) 1 B) 2 C) 3 D) 4 E) 8

```
int T = 83;  
String St;  
St = Integer.toOctalString(T);  
St = St.substring(1,2);  
out.print(St);
```

Question 20

For which value of N below will the code to the right print an odd number?

- A) 200 B) 72 C) 18 D) 44 E) 144

```
int C = 0;  
for(int x=1; x<=N; x++)  
    if(N % x == 0)  
        C++;  
out.print(C);
```


Question 21

In the code to the right, what is output on line #21?

- A) 1 B) 1.5 C) 2 D) 2.5 E) 5

Question 22

In the code to the right, what is output on line #22?

- A) 16 B) 20 C) 24 D) 28 E) 32

Question 23

In the code to the right, what is output on line #23?

- A) 35 B) 40 C) 45 D) 50 E) 55

```
public static int mystery(int M)
{
    if (M > 10)
        return mystery(M-3)+M+3;
    if (M > 5)
        return mystery(M-2)+M+2;
    return M/2;
}
// Client Code
out.println(mystery(3)); //line #21
out.println(mystery(8)); //line #22
out.println(mystery(15)); //line #23
```

Question 24

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

- A) RSITY
B) VERSITY
C) VERSIT
D) TY
E) RIT

```
ArrayList<String>Starsky;
Starsky = new ArrayList<String>();
String Hutch = "UNIVERSITY";
for(int x=0; x<Hutch.length(); x++)
    Starsky.add(Hutch.substring(x));
for(int x=0; x<Starsky.size(); x++)
{
    String A = Starsky.get(x);
    if(A.matches("[AEIOU].*"))
        Starsky.remove(x);
}
out.print(Starsky.get(2));
```

Question 25

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

- A) 1 B) 2 C) 3 D) 4 E) 5

```
int N = 625;
for(int x = 1; x<=3; x++)
    N = (int)(Math.sqrt(N)+ 1);
out.print(N);
```

Question 26

In the client code to the right, what is the output of line #26?

- A) 0
- B) 22
- C) 33
- D) 44
- E) 55

Question 27

In the client code to the right, what is the output of line #27?

- A) 0
- B) 22
- C) 33
- D) 44
- E) 55

Question 28

In the client code to the right, what is the output of line #28?

- A) 0
- B) 22
- C) 33
- D) 44
- E) 55

```
public class State
{
    public int Apple = 44;
    private int Bob = 33;

    public State(int B)
    {
        Bob = B;
    }
    public State()
    {
        Apple = 22;
    }
    public int getA()
    {
        return Apple;
    }
    public int getB()
    {
        return Bob;
    }
}

public class Regional extends State
{
    public int getA()
    {
        return super.getB();
    }
}

public class District extends State
{
    public int getA()
    {
        return super.getA();
    }
}

// Client Code
State Andy = new State(55);
Regional Bette = new Regional();
District Deb = new District();
out.println(Andy.getA()); //line 26
out.println(Bette.getA()); //line 27
out.println(Deb.getA()); //line 28
```

<p>Question 29</p> <p>In the code to the right, what is output on line #29?</p> <p>A) 6 B) 7 C) 8 D) 9 E) 10</p>	<pre>ArrayList<Integer> Yes; ArrayList<Integer> No; ArrayList<Integer> Maybe;</pre>
<p>Question 30</p> <p>In the code to the right, what is output on line #30?</p> <p>A) 6 B) 7 C) 8 D) 9 E) 10</p>	<pre>Yes = new ArrayList<Integer>(); No = new ArrayList<Integer>(); Maybe = new ArrayList<Integer>();</pre>
<p>Question 31</p> <p>In the code to the right, what is output on line #31?</p> <p>A) 14 B) 15 C) 16 D) 17 E) 18</p>	<pre>for(int x=0; x<=50; x+=7) Yes.add(x); for(int x=0; x<=50; x+=5) if(!Yes.contains(x)) No.add(x); for(int x=0; x<=50; x+=2) if(!Yes.contains(x)&&!No.contains(x)) Maybe.add(x); out.println(Yes.size()); //Line #29 out.println(No.size()); //Line #30 out.println(Maybe.size()); //Line #31</pre>
<p>Question 32</p> <p>What is the output of the code segment to the right?</p> <p>A) 15 B) 18 C) 27 D) 36 E) 45</p>	<pre>int C = 0; int x = 15; for(; x<100;) { if(x/10==6) C += x%10; x+=3; } out.print(C);</pre>
<p>Question 33</p> <p>What is the output of the code segment to the right?</p> <p>A) AEIMQUZ B) AAEIMQUY C) AAEJNRVZ D) AAEIMPTY E) AAZ</p>	<pre>String St = "A"; int C = 0; for(int x=0; x<25; x=x+4) St += " " + (char) (St.charAt(C) + x); out.println(St);</pre>
<p>Question 34</p> <p>What is the output of the code segment to the right?</p> <p>A) 7 B) 12 C) 64 D) 81 E) 93</p>	<pre>int M = 4; int H = 3; int T = 1; for(int x=1; x<=H; x++) T *= M; out.println(T);</pre>

Question 35

In the code to the right, what is output on line #35?

- A) 0
- B) 2
- C) 4
- D) 6
- E) 8

```
Stack<Integer> Up;
Stack<Integer> Down;
Up = new Stack<Integer>();
Down = new Stack<Integer>();
for(int x=12; x<=48; x+=6)
{
    Up.push(x%10);
    Down.push(x/10);
}
out.print(Up.peek()); //Line #35
for(int x=1; x<=4; x++)
{
    Down.push(Up.pop());
}
out.print(Down.peek()); //Line #36

for(int x=1; x<=5; x++)
{
    Up.push(Down.pop());
}
out.print(Up.peek()); //Line #37
```

Question 36

In the code to the right, what is output on line #36?

- A) 0
- B) 2
- C) 4
- D) 6
- E) 8

```
int[][]Box = {{2,3,4,5,6},
              {9,7,5,3,1},
              {8,9,0,1,2},
              {7,6,5,4,3}};

int C = 0;
for(int x=1; x< Box.length-1; x++)
    for(int y=1; y< Box[0].length-1;
        y++)
        C += Box[x][y];
out.print(C);
```

Question 37

In the code to the right, what is output on line #37?

- A) 0
- B) 2
- C) 4
- D) 6
- E) 8

10 9 8 7 6 5 * + - + -

Question 38

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

- A) 25
- B) 32
- C) 45
- D) 78
- E) 95

Question 39

Evaluate the postfix expression to the right.

Write your answer in blank #39.

Question 40

Create a binary search tree inserting the letters "AUSTIN". Start with A and end with N. How many leaves will the tree have? Write your answer in blank #40.

★ ANSWER KEY 2024 State – CONFIDENTIAL ★

Questions (+6 points for each correct answer, -2 points for each incorrect answer)

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

* See "Explanation" section below for alternate, acceptable answers.

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.

This page intentionally left blank

Explanations:

1.	A	$7654_8 = 111\ 110\ 101\ 100$ (8 ones) $AAA_{16} = 1010\ 1010\ 1010$ (6 ones) $32132_4 = 11\ 10\ 01\ 11\ 10$ (7 ones) $444444_8 = 100\ 100\ 100\ 100\ 100\ 100$ (6 ones) $63_{10} = 111111$ (6 ones)																																
2.	B	$22 \% 30 * 4 / 10$ $22 * 4 / 10$ $88 / 10$ 8																																
3.	D	<code>Alphabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"</code> <code>Z = Alphabet.substring(10,15) = "KLMNO"</code> <code>Z.substring(3) = "NO"</code>																																
4.	A	<code>A = true;</code> <code>B = !A A = true</code> <code>C = !B && A = false</code> <code>B ^ C = true</code>																																
5.	C	$T = 10 + 20 * 30 / 40 = 10 + 600/40 = 25$ $R = 40 + 30 * 20 / 10 = 40 + 600/10 = 100$ $H = T + R >> 2 = 125 >> 2 = 62 >> 1 = 31$																																
6.	C	One cool trick is to multiply the numbers "under the radical" then take the square root. So we are taking the square root of $10*5*2*2*2$ or 400. The square root of 400 is 20.																																
7.	E	<pre>int A = 11; int B = 10; int C = 12; The consequence on the left in each case is attached to the if. The statement on the right is not a consequence, it will happen anyway.</pre> <table style="margin-left: 40px;"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td></td> <td>11</td> <td>10</td> <td>12</td> </tr> <tr> <td><code>if(A>C) A++; B--;</code></td> <td>false 11</td> <td>9</td> <td>12</td> </tr> <tr> <td><code>if(A>B) B++; C--;</code></td> <td>true 11</td> <td>10</td> <td>11</td> </tr> <tr> <td><code>if(B>C) C++; A--;</code></td> <td>false 10</td> <td>10</td> <td>11</td> </tr> <tr> <td><code>if(C>B) A++; B--;</code></td> <td>true 11</td> <td>9</td> <td>11</td> </tr> <tr> <td><code>if(B>A) B++; C--;</code></td> <td>false 11</td> <td>9</td> <td>10</td> </tr> <tr> <td><code>if(C>A) C++; A--;</code></td> <td>false 10</td> <td>9</td> <td>10</td> </tr> </tbody> </table>		A	B	C		11	10	12	<code>if(A>C) A++; B--;</code>	false 11	9	12	<code>if(A>B) B++; C--;</code>	true 11	10	11	<code>if(B>C) C++; A--;</code>	false 10	10	11	<code>if(C>B) A++; B--;</code>	true 11	9	11	<code>if(B>A) B++; C--;</code>	false 11	9	10	<code>if(C>A) C++; A--;</code>	false 10	9	10
	A	B	C																															
	11	10	12																															
<code>if(A>C) A++; B--;</code>	false 11	9	12																															
<code>if(A>B) B++; C--;</code>	true 11	10	11																															
<code>if(B>C) C++; A--;</code>	false 10	10	11																															
<code>if(C>B) A++; B--;</code>	true 11	9	11																															
<code>if(B>A) B++; C--;</code>	false 11	9	10																															
<code>if(C>A) C++; A--;</code>	false 10	9	10																															
8.	B	<p>C starts as 6</p> <p>Going through the loops adds 5, 3, and 0 to C</p> <p>The total is 14</p>																																
9.	C	<p>"ABCDEFGH IJ"</p> <p>With each pass through the loop, we lose an item from the front and from the back.</p> <p>"BCDEFGHI "</p> <p>"CDEFGH"</p> <p>"DEFG"</p>																																
10.	E	<code>(int) 'A' - (int) 'a'</code> $65 - 97 = -32$																																

11.	E	<pre> int A = 4; int B = 7; int C = 8; if(A<B) out.print("SO "); else out.print("VERY "); if(C<B) out.print("TIRED "); else out.print("HUNGRY "); </pre>
12.	B	<pre> A = 30 & 29; 11110 & 11101 = 11100 B = 29 & 28; 11101 & 11100 = 11100 C = A B; 11100 11100 = 11100 = 28 </pre>
13.	A	<pre> A = 12345678; B = A % 10 = 8 C = A / 100 = 123456 D = C % 10 = 6 E = C / 100 = 1234 F = E % 10 = 4 G = E / 100 = 12 B+D+F+G = 8+6+4+12 = 30 </pre>
14.	C	<pre> int x = 100; while (x>75) x=x-6; x stops at 70 while (x<90) x=x+4; x stops at 90 while (x>40) x=x-10; x stops at 40 </pre>
15.	A	Each item in A is compared to each item in B to see if they have the same tens digit. This happens 12 times
16.	A	None of the String objects are the same. Only A and B have the same "letters" 1 is the answer.
17.	D	<pre> double A = 20.237; out.printf("%.2f ", A); Print 20.24 and stay!!! out.print("UNIVERSITY "); Print "UNIVERSITY" then stay!!! out.println("OF TEXAS "); Print "OF TEXAS " then next line!!! out.print("INTERSCHOLASTIC "); Print "INTER... " then stay!!! out.println("LEAGUE "); Print "LEAGUE " then next line!!! </pre>
18.	C	Many numbers join House (no duplicates, mind you) The loop prints the members of the set. Since it is a TreeSet, the numbers are in increasing order. 35 is the last number. 5 is printed.
19.	B	<pre> int T = 83; St = Integer.toOctalString(T); "123" St = St.substring(1,2); "2" </pre>

20.	E	Only perfect squares have odd numbers of factors.
21.	A	mystery(3) = 3/2 = 1
22.	B	mystery(8) = mystery(6) + 10 = 20!!!! mystery(6) = mystery(4) + 8 = 10 mystery(4) = 4/2 = 2
23.	E	mystery(15) = mystery(12) + 18 = 55!!!! mystery(12) = mystery(9) + 15 = 37 mystery(9) = mystery(7) + 11 = 22 mystery(7) = mystery(5) + 9 = 11 mystery(5) = 5/2 = 2
24.	A	This list is created. ["UNIVERSITY", "NIVERSITY", "IVERSITY", "VERSIY", "ERSITY", "RSITY", "SITY", "ITY", "TY", "Y"] Remove any item that starts with one of the 5 vowels. This gives us: ["NIVERSITY", "VERSIY", "RSITY", "SITY", "TY", "Y"] Item #2 is "RSITY"
25.	C	int N = 625; for(int x = 1; x<=3; x++) N = (int)(Math.sqrt(N)+ 1); out.print(N); N = 625 1st Pass: N = 26 2nd Pass: N = 6 (int)(5 point something + 1) 3rd Pass: N = 3 (int)(2 point something + 1)
26.	D	Line #26 - Andy is a State object. He never changes the initial value of Apple since we called the single parameter constructor.
27.	C	
28.	B	
29.	C	Yes contains the 8 multiples of 7 from 0 to 50 [0, 7, 14, 21, 28, 35, 42, 49] The size is 8
30.	D	No contains multiples of 5 from 0 to 50 that are not also multiples of 7. This leaves out 0 and 35. [5, 10, 15, 20, 25, 30, 40, 45, 50] The size is 9.
31.	D	Maybe contains even numbers from 0 to 50 that are neither multiples of 5 nor multiples of 7. This leaves out 0, 10, 14, 20, 28, 30, 40, 42, 50 [2, 4, 6, 8, 12, 16, 18, 22, 24, 26, 32, 34, 36, 38, 44, 46, 48] The size is 17.

32.	B	This for loop acts as a while loop. We will test all multiples of 3 from 3 to 99. If the number is in the 60s (60, 63, 66, 69), we add the ones digits to the total. That total is 18.
33.	B	Starting with "A" we add first, another "A". Then, we add every 4th letter beyond A to the string. "EIMQUY" to get "AAEIMQUY"
34.	C	The calculates M^H which $4^3 = 64$
35.	E	After the first loop:
36.	A	Up = [2,8,4,0,6,2,8]
37.	C	Down = [1,1,2,3,3,4,4] Peek at Up, see an 8 After the second loop: Up = [2,8,4] Down = [1,1,2,3,3,4,4,8,2,6,0] We popped 4 from Up, and pushed them to Down. Peek at Down, see a 0. After the third loop: Up = [2,8,4,0,6,2,8,4] Down = [1,1,2,3,3,4] We popped 5 from Down, and pushed them to Up. Peek at Up, see a 4.
38.	A	2 3 4 5 6 9 7 5 3 1 8 9 0 1 2 7 6 5 4 3 The loops accumulate a total for only the interior elements. They omit the first and last rows and the first and last columns. $7+5+3+9+0+1 = 25$
39.	30	10 9 8 7 6 5 * + - + - 10 9 8 7 30 + - + - 10 9 8 37 - + - 10 9 -29 + - 10 -20 - 30
40.	2	After building the tree, N and T are the only two nodes with zero children.