QUESTION 1

What is the sum of the binary numbers 1101_2 + 1001_2?

A. 10110_2 B. 11111_2 C. 1001_2 D. 0100_2 E. 2102_2

QUESTION 2

How many *'s are printed by the loop to the right?

A. 0 B. 9 C. 10

D. 11 E. 45

```
for (int i=0; i<10; ++i)
  System.out.print("*");
```

QUESTION 3

What is output by the code to the right?

A. Zero B. Negative

C. Positive D. Error

E. The code does not compile

```
int x = 4;
switch {
case (x==0): System.out.print("Zero");
             break;
case (x<0):  System.out.print("Negative");
             break;
case (x>0):  System.out.print("Positive");
             break;
default:     System.out.print("Error");
}
```

QUESTION 4

What is output by the code to the right?

A. 1 B. 2 C. 3

D. 4 E. 5

```
int [] intArray = {1, 2, 3, 4, 5};

System.out.print(intArray[3]);
```

QUESTION 5

What is output by the code to the right?

A. 12 B. 9 C. 6

D. 3 E. 0

```
int x = 1, y = 2, z = 3;
++x;
--y;
z *= (x + y);
System.out.print(z);
```

QUESTION 6

Where can the constructor `MyClass()` be called?

A. In any class

B. In any class in the same package as `MyClass`

C. In any class in the same file as `MyClass`

D. Only within `MyClass`

E. In `MyClass` and in its subclasses

```
public class MyClass {
  public MyClass() {
    // code not shown
  }

  // other methods and data not shown
}
```

QUESTION 7

If a method has a parameter of type `A`, which of these can be passed to the method?

A. An object of type `A`

B. An object of type `B`

C. An object of type `C`

D. An object of type `A` or `B`

E. An object of type `A`, `B`, or `C`

```
public class A {
  // methods and data not shown
}

public class B extends A {
  // methods and data not shown
}

public class C extends B {
  // methods and data not shown
}
```

QUESTION 8

What is the value of `sum` after executing the code to the right on the input below?

```
12 3 4 10 0 3 -2 14
```

A. `12` B. `32` C. `30`

D. `29` E. `44`

```
Scanner input = new Scanner(System.in);
int x = 1, sum = 0;
while (x >= 0) {
  x = input.nextInt();
  sum = sum + x;
}
```

QUESTION 9

What replaces **<*1>** in the code to the right to throw an `IllegalArgumentException` when the value passed to the constructor is not positive?

A. `throw Exception(IllegalArgument)`

B. `throw IllegalArgumentException`

C. `throw new Exception(IllegalArgument)`

D. `throw new IllegalArgumentException()`

E. Either A or B

QUESTION 10

What replaces **<*2>** in the code to the right to return the product of the mathematical constant pi and the square of the data member `radius`?

A. `Math.PI * this.radius * this.radius`

B. `Math.PI * this.radius^2`

C. `Math.PI * radius * radius`

D. `Math.PI * radius^2`

E. Either A or C

```
public class Circle {

  public Circle(double r) {
    if (r>0) radius = r;
    else <*1>;
  }

  public double area() {
    return <*2>;
  }

  private double radius;
}
```

QUESTION 11

Which of these expressions evaluates to a `String` containing only a lower case `"u"`?

A. `s.substring(0,1).toLowerCase()`

B. `s[0].toLowerCase`

C. `Character.toLowerCase(s.charAt(0))`

D. `s.toLowerCase().substring(1,1)`

E. `s.charAt(1).toLowerCase()`

```
String s = "UIL";
```

QUESTION 12

What matrix is returned by the static method call `process(m)` where `m` is the matrix below?

1	2	3
4	5	6
7	8	9

A.

1	2	3
4	5	6
7	8	9

B.

33	30	27
33	30	27
33	30	27

C.

28	26	24
22	20	18
16	14	12

D.

28	24	20
24	20	16
20	16	12

E.

0	0	0
0	0	0
0	0	0

```
public static int help1(int i) {
  if (i==0) return 1;
  if (i==1 || i==2) return 0;
  return -1;
}

public static int help2(int i) {
  if (i==0 || i==1) return 2;
  if (i==2) return 1;
  return -1;
}

public static int[][] process(int[][] m) {
  int [][] n = {{0,0,0},{0,0,0},{0,0,0}};
  if (m.length!=3) return n;
  for (int i=0; i<3; ++i)
    if (m[i].length!=3) return n;
  for (int i=0; i<3; ++i)
    for (int j=0; j<3; ++j)
      n[i][j] = m[help1(i)][help1(j)] +
                m[help1(i)][help2(j)] +
                m[help2(i)][help1(j)] +
                m[help2(i)][help2(j)];
  return n;
}
```

QUESTION 13

Which of these regular expressions matches both `"aggie"` and `"longhorn"`, but not `"sooner"`?

A. `[als].*[enr]`

B. `.*g.*`

C. `.....(orn)+`

D. `[^sooner]`

E. More than one of these

QUESTION 14

Both `fib1()` and `fib2()` compute a value from the Fibonacci sequence. What is returned by `fib1(7)` (and `fib2(7)`)?

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

D. `8` E. `13`

QUESTION 15

What is the running time of `fib1()`? Choose the most restrictive correct answer.

A. `O(1)` B. `O(log n)` C. $O(n^2)$

D. `O(n)` E. `O(n log n)`

QUESTION 16

During the computation of `fib2(n)`, how many times is `fib2()` called with parameter `(n-3)`? (Assume `n` is at least `3`.)

A. `0` B. `1` C. `2`

D. `3` E. `4`

```
public static int fib1(int n) {
  if (n<0) return -1;
  if (n==0) return 0;
  int n0 = 0, n1 = 1;
  while (--n > 0) {
    int temp = n1;
    n1 = n0 + n1;
    n0 = temp;
  }
  return n1;
}

public static int fib2(int n) {
  if (n<0) return -1;
  if (n==0) return 0;
  if (n==1) return 1;
  return fib2(n-1) + fib2(n-2);
}
```

QUESTION 17

Which of these is a possible output of the code to the right?

A. `0` B. `5` C. `10`

D. Both A and B E. Both B and C

```
Random r = new Random();

System.out.print(r.nextInt(10));
```

QUESTION 18

Which package contains the `Collection` interface and related classes like `LinkedList` and `HashSet`?

A. `java.util` B. `java.lang` C. `java.io`

D. `java.collection` E. `java.container`

QUESTION 19

What is output by the code to the right?

A. `5` B. `10` C. `15`

D. `1015` E. `105`

```
int x = 10, y = 5, z = 15;

System.out.print((x<y)?((y<z)?z:y):
                 ((x<z)?x:z));
```

QUESTION 20

Which of these constuctors for class `Two` sets `x` to `a`, `y` to `b`, and `z` to `c`?

A.
```
public Two(int a, int b, int c) {
  x = a;  y = b;  z = c;
}
```

B.
```
public Two(int a, int b, int c) {
  super(a);  y = b;  z = c;
}
```

C.
```
public Two(int a, int b, int c) {
  super.x = a;  y = b; z = c;
}
```

D. Both A and C are correct

E. No such constructor is possible

QUESTION 21

What is output by the code below?

```
One o = new One(17);
o.y = 14;
System.out.print(o.getX());
```

A. `14` B. `17`

C. `1417` D. `17 14`

E. The code does not compile

```
public class One {
  public One(int a) { x = a; }
  public int getX() { return x; }
  private int x;
  public int y;
}

public class Two extends One {
  // constructor in question
  public int getZ() { return z; }
  private int z;
}
```

QUESTION 22

What does static method `mystery()` do?

A. Converts a string representing a positive binary number to an integer

B. Converts a string representing a positive binary number to a hexadecimal number

C. Converts a string representing a positive hexadecimal number to an octal number

D. Converts a string representing a positive hexadecimal number to an integer

E. Converts a string representing a positive hexadecimal number in reverse order to an integer

QUESTION 23

What is returned by `mystery("20")`?

A. `32` B. `20`

C. `"CF"` D. `2`

E. `"10100"`

```
public static int mystery(String s) {
  int n = 0;
  for (int i=0; i<s.length(); ++i) {
    n = n * 16;
    if (s.charAt(i) >= '0' &&
        s.charAt(i) <= '9')
      n = n + (s.charAt(i) - '0');
    else if (s.charAt(i) >= 'A' &&
             s.charAt(i) <= 'F')
      n = n + 10 + (s.charAt(i) - 'A');
    else return 0;
  }
  return n;
}
```

QUESTION 24

What is returned by `modify("testtest")`?

A. `"tsettset"` B. `"testtset"`

C. `"testtest"` D. `"tttttttt"`

E. `"testttt"`

QUESTION 25

What is returned by `modify("testing")`?

A. `"tesgint"` B. `"testing"`

C. `"gnitset"` D. `"testset"`

E. `"ggggggg"`

```
public static String modify(String s) {
  StringBuffer sb = new StringBuffer(s);
  int n = s.length() / 2;
  int i;
  for (i=0; i<n; ++i) {
    char ch = sb.charAt(i);
    sb.setCharAt(i,sb.charAt(n+i));
    sb.setCharAt(n+i, ch);
  }
  for (i=s.length()-1; i>=s.length()-n;
                                    --i) {
    char ch = sb.charAt(i);
    sb.setCharAt(i,sb.charAt(i-n));
    sb.setCharAt(i-n, ch);
  }
  return sb.toString();
}
```

QUESTION 26

Which of these methods could be added to the `Rectangle` class to compute the area of a rectangle?

A.
```
public double area() {
  return length * width;
}
```

B.
```
public float area() {
  return length * width;
}
```

C.
```
public int area() {
  return length * width;
}
```

D.
```
public static float area() {
  return length * width;
}
```

E.
```
public static int area() {
  return length * width;
}
```

QUESTION 27

Which interface must the `Rectangle` class be modified to implement if you wish to store `Rectangle` objects in a `TreeSet` using the default `TreeSet` constructor?

A. `Serializable`

B. `Cloneable`

C. `Closeable`

D. `Comparable`

E. `Customizable`

```
public class Rectangle {

  public Rectangle(double l, double w) {
    length = l; width = w;
  }

  private double length, width;

}
```

QUESTION 28

What are the contents of `a` after the first pass through the outer loop of the static method call `sort(a)` where `a` is the array below?

9	10	-3	-2	19	8

A.

-3	-2	8	9	10	19

B.

-2	-3	8	9	10	19

C.

9	10	-3	-2	19	8

D.

9	-3	-2	10	8	19

E.

-3	10	9	-2	19	8

```
public static void sort(int [] a) {
  int min, minindex;
  for (int i=0; i<a.length-1; ++i) {
    min = a[i];
    minindex = i;
    for (int j=i+1; j<a.length; ++j)
      if (a[j]<a[minindex]) {
        min = a[j];
        minindex = j;
      }
    a[minindex] = a[i];
    a[i] = min;
  }
}
```

QUESTION 29

Which sorting algorithm is being implemented?

A. Quick sort
B. Merge sort
C. Selection sort
D. Insertion sort
E. None of these

QUESTION 30

Assume the `MyQueue` class is implemented correctly and that the `dequeue()` method returns the item being dequeued. What is output by the code to the right?

A. `Thisisa test`
B. `a testisThis`
C. `Thisisatest`
D. `sihTsitset a`
E. `tset asisihT`

```
MyQueue<String> q = new MyQueue<String>;

q.enqueue("This");
q.enqueue("is");
q.enqueue("a test");

System.out.print(q.dequeue());
System.out.print(q.dequeue());
System.out.print(q.dequeue());
```

QUESTION 31

What replaces **<*1>** in the code to the right to cause `a` to take on all of the values in `array` in order?

A. `for (a in array)`
B. `for (array : int a)`
C. `for (int a : array)`
D. `for ((int)a : (int [])array)`
E. `for ( array / int a)`

```
int [] array;

// code not shown to initialize array

int max = Integer.MIN_VALUE;

<*1> {
  if (a > max) max = a;
}
```

QUESTION 32

What replaces **<*1>** in the code to the right with the name of the method called when a `Food` object is passed to `System.out.print()`?

A. `toString()`

B. `print()`

C. `output()`

D. `stringify()`

E. `showName()`

QUESTION 33

What interface name replaces **<*2>** in the code to the right to iterate through all the `Food` in a `Recipe`?

A. `Set<Food>` B. `Pair`

C. `Double` D. `Map.Entry`

E. `Iterator`

QUESTION 34

Assume **<*1>** and **<*2>** are filled in correctly. Which of these best describes the return value of the `mystery()` static method?

A. All of the instructions in the recipes

B. The total weight of all food needed

C. The weight of food needed for each recipe with the weights of food within a recipe added together

D. The weight of each food needed in any recipe with the ingredients from each recipe listed separately

E. The weight of each food needed in any recipes with the weights of any food used in multiple recipes combined into one entry

```
public class Food {
  public Food(String name) {
    this.name = name;
  }

  public String <*1> { return name; }

  private String name;

  // other methods and data not shown
}

public class Recipe {
  // constructors and methods not shown

  private Map<Food, Double> ingredients;

  // ingredients holds the type of each
  // ingredient and the weight needed

  private List<String> instructions;

  public static Map<Food, Double>
          mystery(List<Recipe> l) {
    Map<Food, Double> m =
                new HashMap<Food, Double>();
    for (Recipe r : l) {
      for (<*2><Food, Double> item :
                 r.ingredients.entrySet()) {
        if (m.containsKey(item.getKey())) {
          double d = m.get(item.getKey());
          d += item.getValue();
          m.put(item.getKey(), d);
        }
        else
          m.put(item.getKey(),
                           item.getValue());
      }
    }
    return m;
  }
}
```

QUESTION 35

Which of these sorts the elements of `intArray`?

A. `intArray.sort()`

B. `intArray.sort(intArray.length)`

C. `Arrays.sort(intArray)`

D. `Array.sort(intArray())`

E. `Array.sort(intArray,`
`intArray.length)`

```
int [] intArray;

//code to initialize intArray not shown
```

QUESTION 36

What is the value of `e3`?

A. `6` B. `5` C. `4`

D. `3` E. `2`

QUESTION 37

What is the value of `e4`?

A. `2` B. `1` C. `0`

D. `-1` E. `-2`

```
double e = 2.718281828;

int e1 = (int)e;

int e2 = (int)Math.floor(e);

int e3 = (int)Math.max(e1, e2);

int e4 = (int)Math.ceil(e3 - e);
```

QUESTION 38

What replaces **<*1>** to check whether the string read is `"test"`?

A. `s == "test"`
B. `s = "test"`
C. `s.equals("test")`
D. `s.compareTo("test") == 0`
E. Either C or D

QUESTION 39

Assume **<*1>** is filled in correctly. Which of these best describes the `String` returned by `next()`?

A. The first non-whitespace character
B. Starts at the first non-whitespace character and ends with the last character before the next whitespace character
C. Starts at the first non-whitespace character and ends at the end of the line
D. Starts at the first character and ends at the end of the line
E. All of the current input

```
Scanner input = new Scanner(System.in);

String s = input.next();

if (<*1>) System.out.print("Match");
```

QUESTION 40

Which of these data structures allows access to any element in constant time?

A. linked list B. array C. priority queue

D. binary search tree E. heap

Computer Science Answer Key
UIL Invitational A 2006

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