Note: Correct responses are based on Java, J2sdk v 6.0, from Sun Microsystems, Inc. All provided code segments are intended to be syntactically correct, unless otherwise stated (i. e. `error` is an answer choice) and any necessary Java 2 Standard Packages have been imported. Ignore any typographical errors and assume any undefined variables are defined as used.

QUESTION 1

What is 81_{11} plus 28_{11} ?

A. 516_4 B. 91_{10} C. 109_{11} D. $A9_{11}$ E. 332_7

QUESTION 2

What is output by the code to the right?

A. `3.0` B. `9.0` C. `6.0` D. `0.0`

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

```
double a = 6;
a = a - 9 + a;
System.out.println(a);
```

QUESTION 3

What is output by the code to the right?

A. `50` B. `-25` C. `25` D. `100`

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

```
int b = 50;
b *= 2;
System.out.println(b);
```

QUESTION 4

What is output by the code to the right?

A. `123` B. `200` C. `231` D. `162` E. `141`

```
int much = 0;
for(int c=1; c<21; c=c+2)
  much = much + c + c;
System.out.print(much);
```

QUESTION 5

What is output by the code to the right?

A. `regional` B. `UIL` C. `rocks`

D. `UIL_rocks` E. `regional_UIL`

```
String d = "regional_UIL_rocks";
Integer loc = d.indexOf("_");
System.out.print(d.substring(0,loc));
```

QUESTION 6

What is output by the code to the right?

A. `8` B. `9` C. `11` D. `14` E. `15`

```
long[] ray = {1,5,6,3,2,4,8,9};
Arrays.sort(ray);
System.out.println(ray[4] * ray[2]);
```

QUESTION 7

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

```
p && (p || q)
```

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

QUESTION 8

What is output by the code to the right?

A. `22`

B. `111`

C. `14`

D. `13`

E. `0`

```
int frog = 111;
switch(frog){
   case 111 : frog = 22;
   case 22 : frog = 111;
   default : frog = 22;
}
if(frog > 10)
   frog = 14;
else
   frog = 13;
System.out.println(frog);
```

QUESTION 9

Which of the following could replace **<*1>** in the code of class `TootsieRoll` to the right so that method `bumpUp` would round the price to the nearest whole number?

A. `price = Math.round(price);`

B. `price = Math.floor(price);`

C. `price = Math.ceil(price);`

D. A and B only.

E. A, B, and C.

QUESTION 10

Assuming that **<*1>** is filled correctly, what is output by the line marked `//1` in the code to the right?

A. `4.0`

B. `4`

C. `3.0`

D. `3.5`

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

```
public class TootsieRoll
{
  private double price;

  public TootsieRoll(double p){
    price = p;
  }

  public void bumpUp(){
    <*1>
  }

  public double getPrice(){
    return price;
  }
}

//////////////////////////////////////////
//client code
TootsieRoll toot = new TootsieRoll(3.1);
toot.bumpUp();
System.out.println(toot.getPrice());   //1
```

QUESTION 11

What is output by the code to the right?

A. `9` B. `7` C. `3` D. `11` E. `8`

```
System.out.println(5 | 7 & 9 ^ 2);
```

QUESTION 12

How many lines of output does the code to right produce?

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

```
System.out.printf("jump\\nup\\\\");
```

QUESTION 13

What is output by the code to the right?

A. `7.0` B. `8.0` C. `7` D. `8` E. `9.0`

```
Double big = Math.ceil(7.8);
System.out.println(big);
```

QUESTION 14

Which of the following could replace **<*1>** in method `doIt` in the code to the right?

A. `Integer[]` B. `int[]` C. `double[]`

D. A and B only E. A, B, and C

QUESTION 15

Which of the following could replace **<*2>** in method `doIt` in the code to the right?

A. `answer.intValue();` B. `answer;`

C. `(int)answer;` D. A and B only

E. A, B, and C

QUESTION 16

Assuming that **<*1>** and **<*2>** are filled correctly, what is output by the line marked `//1` in the code to the right?

A. `11` B. `22` C. `13` D. `12` E. `15`

```
public static int doIt( <*1> what)
{
  Integer answer = 0;
  answer += what[0] + 2;
  answer += what[1] + 2;
  answer += what[2] + 2;
  return <*2>
}

//////////////////////////////////////////
//client code

Integer[] how = {1,2,3,4,5};
System.out.println(doIt(how));   //1
```

QUESTION 17

Which of the following could ***NOT*** replace **<*1>** in the code to the right to correctly define variable `storage`?

A. `Integer` B. `float` C. `Long`
D. `double` E. More than one of these.

```
<*1> storage = 65536;
```

QUESTION 18

What is output by the code to the right?

A. `false` B. `true` C. `stop` D. `0` E. `1`

```
boolean k=true, m=true, p=false;
out.println(k && (m || p) ^ k);
```

QUESTION 19

What is output by the code to the right?

A. `[1, 3, 5, 7, 9]`
B. `[4, 8, 1, 3, 5, 7, 9]`
C. `[2, 4, 6, 8]`
D. `[2, 6, 1, 3, 5, 7, 9]`
E. There is no output due to a runtime error.

```
Integer[] z = {2,4,6,8,1,3,5,7,9};
List iList = Arrays.asList(z);
ArrayList<Integer> n;
n = new ArrayList<Integer>(iList);

for(int i=0; i<n.size(); i++)
  if(n.get(i)%2==0)
    n.remove(i);
System.out.println(n);
```

QUESTION 20

What is output by the code to the right?

A. `4.0` B. `0.0`
C. `16.0` D. `32.0`
E. `4.0`

```
double dbl = 32>>2>>2<<2<<2;
System.out.println(dbl);
```

QUESTION 21

Which of the following could replace **<*1>** in the code to the right in order to correctly refer `longSet` to a set object?

A. `new Set();`
B. `new TreeSet<Long>();`
C. `new Set<Long>();`
D. `new HashSet<Long>();`
E. more than one of these

```
SortedSet<Long> longSet = <*1>
```

QUESTION 22

What is returned by the method call `what(new int[]{3,4,5,6,7,8,9,90,0})` ?

A. `1` B. `6` C. `5` D. `2` E. `7`

QUESTION 23

What is returned by the method call `what(new int[]{10,1,5,2,7,13,9,18})` ?

A. `1` B. `6` C. `5` D. `2` E. `7`

```
public static int what(int[] x)
{
  int back=0;
  int old = x[x.length/2];
  for(int it : x)
  {
    if(it>=old)
      ++back;
    if(it<=old)
      --back;
    old = it;
  }
  return back;
}
```

QUESTION 24

What is output by the line marked `//1` in the client code to the right?

A. `2`
B. `9`
C. `0`
D. `3`
E. There is no output due to a syntax error.

QUESTION 25

What is output by the line marked `//2` in the client code to the right?

A. `2`
B. `9`
C. `0`
D. `3`
E. There is no output due to a syntax error.

QUESTION 26

What is output by the line marked `//3` in the client code to the right?

A. `2`
B. `9`
C. `0`
D. `3`
E. There is no output due to a syntax error.

QUESTION 27

What is output by the line marked `//4` in the client code to the right?

A. `2`
B. `9`
C. `0`
D. `3`
E. There is no output due to a syntax error.

```
public class Actor
{
  private int val;

  public Actor(int value)
  {
    setSize(value);
  }

  public void setSize(int value)
  {
    val = value;
  }

  public String toString()
  {
    return "" + val;
  }
}

public class ActorFun
{
  public static void go(Actor a)
  {
    a.setSize(3);
  }

  public static void up(Actor a)
  {
    a = new Actor(9);
    System.out.println(a);       //3
  }
}


////////////////////////////////////
//client code

Actor dude = new Actor(2);
System.out.println(dude);        //1
ActorFun.go(dude);
System.out.println(dude);    //2
ActorFun.up(dude);
System.out.println(dude);        //4
```

QUESTION 28

What is the worst case run-time for a binary search tree traversal? Choose the most restrictive answer.

A. $O(1)$ B. $O(N)$ C. $O(Log_2N)$ D. $O(Log_2N*N)$ E. $O(N^2)$

QUESTION 29

What is returned by the method call `wow(6)` ?

A. `29` B. `-6` C. `-10` D. `19` E. `41`

QUESTION 30

What is returned by the method call `wow(11)` ?

A. `29` B. `-6` C. `-10` D. `19` E. `41`

```
public static int wow(int x)
{
  if(x<0)
    return x%2;
  else if(x%2==1)
    return x+1 + wow(x-2);
  else
    return -x;
}
```

QUESTION 31

What is the worst case run-time for a HashSet? Choose the most restrictive answer.

A. O(1) B. O(N) C. $O(Log_2N)$ D. $O(Log_2N*N)$ E. $O(N^2)$

QUESTION 32

What is output by the code to the right?

A. 40
B. 32
C. 4
D. 64
E. 36

```
Integer bitwise = 64>>1|8^4&2;
System.out.println(bitwise);
```

QUESTION 33

Which of the following could replace **<*1>** in the code to the right so that `thing` would be defined correctly?

A. `String`
B. `Map.Entry`
C. `Integer`
D. `Object`
E. more than one of these

QUESTION 34

Assuming that **<*1>** is filled correctly, what is output by the line marked `//1` in the code to the right?

A. 48 B. 50 C. 68 D. 99 E. 49

QUESTION 35

Assuming that **<*1>** is filled correctly, what is output by the line marked `//2` in the code to the right?

A. 48 B. 50 C. 68 D. 99 E. 49

```
Map<String, Integer> map;
map = new TreeMap<String, Integer>();

Queue<Integer> q;
q = new LinkedList<Integer>();

map.put("2", 50);
map.put("A", 65);
map.put("b", 98);
map.put("c", 99);
map.put("D", 68);
map.put("1", 49);


for( <*1> thing : map.entrySet())
   q.add(map.get(thing.getKey()));

Iterator<Integer> it = q.iterator();
out.println(it.next());                    //1
it.next();
it.next();
out.println(it.next());                    //2
```

QUESTION 36

Which of the following replaces **<*1>** in the code to the right so that all elements in `x` will be visited when method `test` is called?

A. `i<x.size()`
B. `i<x.length`
C. `i<x.length-1`
D. A and B only
E. A, B, and C

QUESTION 37

Assuming that **<*1>** is filled correctly, what is returned by method `test`?

A. true is returned when the value is found
B. true is returned when the value is in the array
C. false is returned when the value is found
D. true is returned each time the loop iterates
E. false is returned each time the loop iterates

```
public class What
{
  public boolean test(int[] x, int y)
  {
    for(int i=0;    <*1>     ;i++)
    {
      if(x[i]==y)
        return false;
    }
    return true;
  }
}
```

QUESTION 38

What is output by the line marked `//1` in the client code to the right?

A. `[]`

B. `[a, 7]`

C. `[A, 15]`

D. `[15, A]`

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

QUESTION 39

What is output by the line marked `//2` in the client code to the right?

A. `[]`

B. `[a, 7]`

C. `[A, 15]`

D. `[15, A]`

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

QUESTION 40

What data structure is being created by class `Locker`?

A. a binary search tree

B. a heap

C. a map

D. a hash table

E. a radix heap tree

```
public class Locker
{
   private LinkedList[] stuff;

   public Locker(int size){
      stuff = new LinkedList[size];
      for(int i=0; i<stuff.length; i++)
        stuff[i] = new LinkedList();
   }

   public void add(Object obj, int size)
   {
      int i = obj.hashCode() % size;
      if(stuff[i].contains(obj)==false)
      {
        stuff[i].add(obj);
      }
   }

   public LinkedList get(int ind)
   {
      return stuff[ind];
   }
}

//////////////////////////////////////////
//client code
Locker sto = new Locker(10);
sto.add(3, 10);
sto.add(2, 10);
sto.add(7, 10);
sto.add(9, 10);
sto.add(15, 10);
sto.add(21, 10);
sto.add("a", 10);
sto.add("A", 10);

System.out.println(sto.get(0));  //1
System.out.println(sto.get(5));  //2
```