1. Which line of code puts a random number into a label?

```vbnet```
LblNum(z).Caption = Int(Rnd() * 100) + 1
```vbnet```

2. Which line of code stores the number in the variable array for sorting.

```vbnet```
SortedNums(z) = LblNum(z).Caption
```vbnet```

3. When the sort algorithm finds two numbers out of order, it swaps them.
   a. Which lines of code do this?

```vbnet```
temp = SortedNums(y)
SortedNums(y) = SortedNums(x)
SortedNums(x) = temp
```vbnet```

b. Why is the temp variable used?

One of the values would be lost if a straight swap was performed. When a variable or property is
overwritten, the original value is lost. Temp holds the first value while the second value overwrites the
first. The value in the Temp variable then overwrites the second, thus completing the swap.

4. The event cmdSort_Click sorts the numbers into order. To better understand the algorithm, try manually
   following the code and keeping track of the variables, x, y, smallest and temp, as well as the variable array
   SortedNums(). Use the randomly generated numbers and the grid shown below.

<table>
<thead>
<tr>
<th>SortedNum</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
</tr>
<tr>
<td>----</td>
</tr>
<tr>
<td>Randomly generated numbers:</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

5. Extend the project to sort 20 numbers rather than 10.

   Extend the control arrays to 20 labels each.
   Declare Const Num = 20 in the general section and replace all instances of 9 throughout the code with
   Num -1

6. Extend the project to sort letters between A and Z.

   Use the Ascii character set. Numbers 65 - 90 represent the letters A – Z.
   LblChars(z).Caption = Char(Int(Rnd * 26) + 65)

   Change the variable array declaration to:
   Dim sortedChars(Num-1) as variant (or String *1)

   Change temp and smallest to variant datatypes.
   Dim temp as variant, smallest as variant

7. There are other ways to sort (some are more efficient than others). Can you think of a different way to sort a
   list of numbers?

   One method would be find the smallest number by looking through the whole list, then swapping this
   with the first element. Repeat by sorting the remaining 9 elements etc. until complete. This involves
   less swapping than the method used in this project.