

Padlock Puzzle Solution

I wrote the following JavaScript code to solve the sample puzzle 1. The three numbers were found to be 6,7 and 9 respectively.

```
1. // Step 1: Listing all the given codes with their conditions
2.
3. // One digit is right but in the wrong place
4. let code1 = [1, 4, 7];
5. // One digit is right and, in its place,
6. let code2 = [1, 8, 9];
7. // Two digits are correct but both are in the wrong place
8. let code3 = [9, 6, 4];
9. // All digits are wrong
10. const code4 = [5, 2, 3];
11. // One Digit is right but in the wrong place
12. let code5 = [2, 8, 6];
13.
14. // Step 2: Declaring an array to store the correct codes
15. let correctCode = new Array(3);
16.
17. // Step 3: Loading the codes into an array
18. let codesArr = [code1, code2, code3, code5];
19.
20. // Step 4: Matching the wrong digits of the array stored in variable code4 with all the
    digits of the codes in the array
21. // The wrong digits are set as 'x'
22.
23. for (let i = 0; i < codesArr.length; i++) {
24.   let codeArr = codesArr[i];
25.   for (let j = 0; j < codeArr.length; j++) {
26.     code4.map((digit) => {
27.       if (digit === codesArr[i][j]) {
28.         codesArr[i][j] = 'x';
29.       }
30.     });
31.   }
32. }
33.
34. // Step 5
35. /**Check if the digits in code1 and code2 are in same position
36. If the digits are in same position, the value is set to 'x' as
37. code1 states that the one digit is in its place and code2 state that the correct digit
    is in right but wrong place.
38. If any digits are in same position, then the statements would contradict each other.
    Hence, proving that the digit is false.
39. */
40. for (let i = 0; i < 3; i++) {
41.   if (code1[i] == code2[i]) {
42.     code1[i] = code2[i] = 'x';
43.   }
44. }
45.
46. // Step 6
47. /**Check if the digits in code2 and code5 are in same position
48. If the digits are in same position, the value is set to 'x' as
49. code1 states that the one digit is in its place and code2 state that the correct digit
    is in right but wrong place.
```

```

50. If any digits are in same position, then the statement would contradict. Hence, proving
    that the digit is false.
51. */
52. for (let i = 0; i < 3; i++) {
53.     if (code2[i] == code5[i]) {
54.         code2[i] = code5[i] = 'x';
55.     }
56. }
57.
58. // Step 7
59. /** In code2 i.e., 189, only the digit 9 remains and as we know the condition in code2
    is that one digit is right and, in its place,
60. the digit 9 is a correct digit and its correct position is the 3rd position*/
61. correctCode[2] = code2[2];
62.
63. // Step 8
64. /**
65.  * It is found that the code5 has only one digit remaining which is 6 and the condition
    of code 5 is one digit is right but in the wrong place.
66.  * Hence, we can conclude that '6' is one of the correct digits but we still do not
    know the correct position for it.
67.  * We also now know that the correct position of digit 6 cannot be 3rd position.
68.  */
69.
70. // Step 9
71. /**
72.  * In code3 i.e., 964, the condition states that only two digits are right but both are
    in the wrong place.
73.  * We already know that the 9 and 6 are among the correct digits.
74.  * Here, since from the condition we know that 6 is in wrong position, the 2nd position
    is wrong for the digit '6'.
75.  * From the previous analogy we also know that the digit 6 cannot be in the 3rd
    position.
76.  * Hence, it is concluded that the correct position for 6 is in the 1st position.
77.  */
78.
79. correctCode[0] = code3[1];
80.
81. // Step 10
82. //From the above analogy, we know that 4 is a wrong digit.
83.
84. // Step 11
85. /**
86.  * Now, we know that the correct digits for 1st and 3rd position.
87.  * In code1 i.e., 147, the condition is that one digit is right but in the wrong place.
88.  *
89.  * We have already analyzed that the digit 4 is wrong digit. And, only 2nd place is
    available.
90.  *
91.  * Hence, the only possible digit left is 4 and only 2nd place is available to place
    it.
92.  * Hence, 4 is the correct digit with the position in 2nd place
93.  *
94.  */
95.
96. correctCode[1] = code1[1];
97.
98. // Step 12
99. // Outputting the correct code which is 679
100. console.log('Correct Code = ' + correctCode);
101.

```

I wrote the following JavaScript code to solve the sample puzzle 2. The three numbers were found to be 0,4 and 2 respectively.

```
1. // Listing all the given codes with their conditions
2.
3. // One digit is right and, in its place,
4. let code1 = [6, 8, 2];
5. // One digit is right but in the wrong place
6. let code2 = [6, 1, 4];
7. // Two digits are right but both are in the wrong place
8. let code3 = [2, 0, 6];
9. // All digits are wrong
10. const code4 = [7, 3, 8];
11. // One Digit is right but in teh wrong place
12. let code5 = [3, 8, 0];
13.
14. // Declaring an array to store the correct codes
15. let correctCode = new Array(3);
16.
17. // Loading the codes into an array
18. let codesArr = [code1, code2, code3, code5];
19.
20. // Matching the wrong digits with all the digits of the codes in the array
21. // The wrong digits are set as 'x'
22.
23. for (let i = 0; i < codesArr.length; i++) {
24.   let codeArr = codesArr[i];
25.   for (let j = 0; j < codeArr.length; j++) {
26.     code4.map((digit) => {
27.       if (digit == codesArr[i][j]) {
28.         codesArr[i][j] = 'x';
29.       }
30.     });
31.   }
32. }
33.
34. /**Check if the digits in code1 and code2 are in same position
35. If the digits are in same position, the value is set to 'x' as
36. code1 states that the one digit is in its place and code2 state that the correct digit
   is in right but wrong place.
37. If any digits are in same position, then the statement would contradict. Hence, proving
   that the digit is false.
38. */
39. for (let i = 0; i < 3; i++) {
40.   if (code1[i] == code2[i]) {
41.     code1[i] = code2[i] = 'x';
42.   }
43. }
44.
45. /** In code1 i.e., 682, only the digit 2 remains and as we know the condition in code1
   is that one digit is right and, in its place,,
46. the digit 2 is a correct digit and its correct position is the 3rd position*/
47. correctCode[2] = code1[2];
48.
49. /**
50. * It is found that the code5 has only one digit remaining which is 0 and the condition
   of code 5 is one digit is right but in the wrong place.
```

```
51. * Hence, we can conclude that '0' is one of the correct digits but we still do not
    know the correct position for it.
52. * We also now know that the correct position of digit 0 cannot be 3rd position.
53. */
54.
55. /**
56. * In code3 that is 206, the condition states that only two digits are right but both
    are in the wrong place.
57. * We already know that the 0 is one of the correct digits.
58. * Here, since from the condition we know that 0 is in wrong position, the 2nd position
    is wrong for the digit 0.
59. * From the previous analogy we also know that the digit 0 cannot be in the 3rd
    position.
60. * Hence, it is concluded that the correct position for 0 is in the 1st position.
61. */
62.
63. correctCode[0] = code3[1];
64.
65. /**
66. * Now, we know that the correct digits for 1st and 3rd position.
67. * In code2 i.e., 614, the condition is that one digit is right but in the wrong place.
68. *
69. * We have already analyzed that the digit 6 is wrong digit.
70. *
71. * And, only 2nd place is available. And since, 1 cannot be the digit as it is in 2nd
    place already and the condition states that the correct digit is the wrong place,
72. * the only possible digit left is 4.
73. * Hence, 4 is the correct digit with the position in 2nd place
74. *
75. */
76.
77. correctCode[1] = code2[2];
78.
79. // Outputting the correct code
80. console.log('Correct Code = ' + correctCode);
81.
```