Challenge September-2019
Crack the Code

A solution with Python  by Jack Jansonius – 8 September 2019

Hints for the September-challenge:
682 – one number is correct and in the correct position
645 – one number is correct but in the wrong position
206 – two numbers are correct but in the wrong positions
738 – nothing is correct
780 – one number is correct but in the wrong position.

# Imperative style:
def checknr(number, checknumber, nr_correct, nr_correct_position):
count_nr_correct = 0
count_nr_correct_position = 0
inumber = str(number).zfill(len(checknumber))  # make string with leading zeros
for i in range(len(checknumber)):
    if inumber[i] in checknumber:
        count_nr_correct += 1
    if inumber[i] == checknumber[i]:
        count_nr_correct_position += 1
return nr_correct = count_nr_correct and \
    nr_correct_position = count_nr_correct_position

# Pythonic style (comprehension):
def checknr2(number, checknumber, nr_correct, nr_correct_position):
inumber = str(number).zfill(len(checknumber))  # make string with leading zeros
return nr_correct == sum(inumber[i] in checknumber for i in range(len(checknumber))) and \
    nr_correct_position == sum(inumber[i]==checknumber[i] for i in range(len(checknumber)))

for cnumber in range (1000):
    if checknr(cnumber, '682', 1, 1) and \
        checknr(cnumber, '645', 1, 0) and \
        checknr(cnumber, '206', 2, 0) and \
        checknr(cnumber, '738', 0, 0) and \
        checknr(cnumber, '700', 1, 0):
        print("Code 1a found: " + str(cnumber).zfill(3))

for cnumber in range (1000):
    if checknr2(cnumber, '682', 1, 1) and \
        checknr2(cnumber, '645', 1, 0) and \
        checknr2(cnumber, '206', 2, 0) and \
        checknr2(cnumber, '738', 0, 0) and \
        checknr2(cnumber, '700', 1, 0):
        print("Code 1b found: " + str(cnumber).zfill(3))

Code 1a found: 052
Code 1b found: 052

Process finished with exit code 0
Another 3-digit crack the code puzzle.

A Number Lock has a 3-Digit Key  
Hints :  
6 3 1 - One number is correct and well placed.  
7 3 0 - Nothing is correct.  
1 0 2 - Two numbers are correct but wrongly placed.  
6 7 8 - One number is correct but wrongly placed.  
0 8 7 - One number is correct and well placed.  

Logic (copied from the website):  
From 1st , 2nd and 3rd clue we know that 1 & 2 are two of the three numbers with 1 being in the last position of the code.  
Now from 4th clue we get to know that 8 is the third number of the code.  
From 5th clue we get that 8 is the 2nd position of code.  
So the number is 281.

```python
# Imperative style:
def checknr(number, checknumber, nr_correct, nr_correct_position):
    count_nr_correct = 0
    count_nr_correct_position = 0
    lnumber = str(number).zfill(len(checknumber))  
    # make string with leading zeros
    for i in range(len(checknumber)):
        if lnumber[i] in checknumber:
            count_nr_correct += 1
        if lnumber[i] == checknumber[i]:
            count_nr_correct_position += 1
    return nr_correct == count_nr_correct and \
    nr_correct_position == count_nr_correct_position

for cnnumber in range(1000):
    if checknr(cnnumber, '631', 1, 1) and \
    checknr(cnnumber, '730', 0, 0) and \
    checknr(cnnumber, '102', 2, 0) and \
    checknr(cnnumber, '678', 1, 0) and \
    checknr(cnnumber, '087', 1, 1):
        print("Code 2 found: " + str(cnnumber).zfill(3))

Code 2 found: 281
```

Process finished with exit code 0
Another 5-digit crack the code puzzle.

Hints for the crack the code cipher puzzle:
79314 – one number is correct but in the wrong position
95643 – two numbers are correct but only one in the right position
57319 – two numbers are correct and in the right position
Sum of the numbers is equal the last 2 numbers (A + B + C + D + E = D*10 + E)
https://brainyyou.com/can-you-crack-this-code-to-open-mobile-phone-puzzle-id-oj1a7/

# Pythonic style (comprehension):

```python
def checknr2(number, checknumber, nr_correct, nr_correct_position):
    lnumber = str(number).zfill(len(checknumber))  # make string with leading zeros
    nr_correct = sum([int(i) for i in checknumber for i in range(len(checknumber))])
    nr_correct_position = sum([int(i) == checknumber[i] for i in range(len(checknumber))])

for cnumber in range(100000):
    if checknr2(cnumber, '79314', 1, 0) and \
    checknr2(cnumber, '95643', 2, 1) and \
    checknr2(cnumber, '57319', 2, 2) and \
    sum(int(i) for i in list(str(cnumber))) == cnumber & 100:
        print('Code 3 found: ' + str(cnumber).zfill(3))
```

Code 3 found: 57620
Code 3 found: 57622
Code 3 found: 57628

Process finished with exit code 0