from math import sqrt; from itertools import count, islice

def patolli(n):
    squares = [9, 16, 25, 36, 49, 64, 81, 100]
    row1 = [1, 52]
    row2 = [2, 51]
    row3 = [7, 6, 4, 3, 50, 49, 48, 47, 46]
    row4 = [8, 9, 10, 11, 12, 41, 42, 43, 44, 45]
    row5 = [17, 16, 15, 14, 13, 40, 39, 38, 37, 36]
    row6 = [18, 19, 20, 21, 22, 31, 32, 33, 34, 35]
    row7 = [23, 30]
    row8 = [24, 29]
    row9 = [25, 28]
    row10 = [26, 27]
    allrows = [row1, row2, row3, row4, row5, row6, row7, row8, row9, row10]
    column1 = [7, 8, 17, 18]
    column2 = [6, 9, 16, 19]
    column3 = [5, 10, 15, 20]
    column4 = [4, 11, 14, 21]
    column5 = [1, 2, 3, 12, 13, 22, 23, 24, 25, 26]
    column6 = [52, 51, 50, 41, 40, 31, 30, 29, 28, 27]
    column7 = [49, 42, 39, 32]
    column8 = [48, 43, 38, 33]
    column9 = [47, 44, 37, 34]
    column10 = [46, 45, 36, 35]
    allcols = [column1, column2, column3, column4, column5, column6, column7, column8, column9, column10]
    list_1 = n.split(' ')
    occupied = []
    occupied.extend((int(list_1[0]), int(list_1[1]), int(list_1[2])))
    list_1 = list_1[3:]
    num_of_rolls = int(list_1[0])
    list_1 = list_1[1:]
    dierolls = []
    for i in list_1:
        dierolls.append(int(i))
    for marker in dierolls:
        current_spot = current_spot + marker
        if (current_spot in occupied):
            current_spot = current_spot - marker
            if current_spot > 52:
                current_spot = current_spot - marker
                if (current_spot > 1 and all(current_spot % k for k in islice(count(2), int(sqrt(current_spot)-1)) or
                    if occupied[0] > current_spot and occupied[0] <= (current_spot + 6):
                        current_spot = current_spot + (occupied[0] - current_spot - 1)
                        current_spot = current_spot + (occupied[1] - current_spot - 1)
                        current_spot = current_spot + (occupied[2] - current_spot - 1)
                    else:
                        current_spot = current_spot + 6
                    elif current_spot in squares:
                        if occupied[0] < current_spot and occupied[0] >= (current_spot - 6):
                            current_spot = current_spot + (occupied[0] - current_spot + 1)
                            current_spot = current_spot + (occupied[1] - current_spot + 1)
                            current_spot = current_spot + (occupied[2] - current_spot + 1)
                        else:
                            current_spot = current_spot - 6
                    else:
                        current_spot = current_spot - 6
                        elif current_spot > 1 and all(current_spot % k for k in islice(count(2), int(sqrt(current_spot)-1)))) == False:
                            checkup = []
                            list_of_rows = []
                            list_of_columns = []
for step in range(current_spot - marker, current_spot + 1):
    checkup.append(step)
for number in checkup:
    for rowl in allrows:
        if number in rowl:
            list_of_rows.append(rowl)
    for coll in allcols:
        if number in coll:
            list_of_columns.append(coll)
for x in list_of_rows:
    list_of_rows = [1 if x == row1 else x for x in list_of_rows]
    list_of_rows = [2 if x == row2 else x for x in list_of_rows]
    list_of_rows = [3 if x == row3 else x for x in list_of_rows]
    list_of_rows = [4 if x == row4 else x for x in list_of_rows]
    list_of_rows = [5 if x == row5 else x for x in list_of_rows]
    list_of_rows = [6 if x == row6 else x for x in list_of_rows]
    list_of_rows = [7 if x == row7 else x for x in list_of_rows]
    list_of_rows = [8 if x == row8 else x for x in list_of_rows]
    list_of_rows = [9 if x == row9 else x for x in list_of_rows]
    list_of_rows = [10 if x == row10 else x for x in list_of_rows]
for y in list_of_columns:
    list_of_columns = [1 if y == column1 else y for y in list_of_columns]
    list_of_columns = [2 if y == column2 else y for y in list_of_columns]
    list_of_columns = [3 if y == column3 else y for y in list_of_columns]
    list_of_columns = [4 if y == column4 else y for y in list_of_columns]
    list_of_columns = [5 if y == column5 else y for y in list_of_columns]
    list_of_columns = [6 if y == column6 else y for y in list_of_columns]
    list_of_columns = [7 if y == column7 else y for y in list_of_columns]
    list_of_columns = [8 if y == column8 else y for y in list_of_columns]
    list_of_columns = [9 if y == column9 else y for y in list_of_columns]
    list_of_columns = [10 if y == column10 else y for y in list_of_columns]
multiples = []
for mult in range(1, 52):
    if mult not in occupied:
        if (mult * marker) > (current_spot - marker):
            multiples.append(mult * marker)
        a = ''
normal_spot = current_spot
count1 = 1
count2 = 2
while count1 != len(list_of_columns) and count2 != len(list_of_rows):
    if list_of_columns[count1] != list_of_columns[count1 - 1]:
        if list_of_rows[count2] != list_of_rows[count2 - 1]:
            a = True
            current_spot = multiples[0]
            if current_spot in occupied:
                current_spot = normal_spot - marker
                break
        else:
            count1 = count1 + 1
            count2 = count2 + 1
    else:
        count1 = count1 + 1
        count2 = count2 + 1

if current_spot == 52:
    return "GAME OVER"
else:
    return current_spot

print(patolli(input('Input:')))
print(patolli(input('Input:')))
print(patolli(input('Input:')))