

## Challenge 20 - forfor

[« Prev](#)   [Next »](#)

You are responsible for managing the supercomputing data center at the Technology United Enterprise New Template Institute. However, since you discovered crypto-currencies, you have started mining litecoin with it, subsequently burning all the hardware and leaving the machinery useless.

Your boss has asked you to give him the result of the following function for a list of numbers. Since you cannot use the supercomputer's power, you will have to solve it using your personal computer.

```
long long calc(int n) {
    long long ret = 0,a,b,c,d,e,f,g,M = 3211123;

    for(a = 1; a <= n; a++)
        for(b = 1; b <= n; b++)
            for(c = 1; c <= n; c++)
                for(d = 1; d <= n; d++)
                    for(e=1; e <= n; e++)
                        for(f=1; f <= n; f++)
                            for(g=1; g <= n; g++)
                                if(a + b + c + d + e + f + g == n)
                                    ret = (ret + a*a + b*b + c*c + d*d +
e*e + f*f + g*g) % M;
    return ret;
}
```

### Input

The input will consist of integers  $1 \leq n \leq 10^9$  each on a different line.

## Output

You need to provide the sum of all the values for each of the given  $n$ , modulo 3211123.

## Sample input

```
5
7
8
61
```

Sample output

```
98077
```

## Submit & test your code

To test and submit code we provide a set of tools to help you. Download [contest tools](#) if you haven't already done that. You will then be able to test your solution to this challenge with the challenge tokens.

```
challenge tokens: CHALLENGE_20, CHALLENGE_SUBMIT_20
```

### To test your program

```
./test_challenge CHALLENGE_20 path/program
```

A nice output will tell you if your program got the right solution or not. You can try as many times as you need.

### To test your program against the input provided in the submit phase

```
./test_challenge CHALLENGE_SUBMIT_20 path/program
```

During the submit phase, in some problems, we might give your program

harder inputs. As with the test token, a nice output will tell you if your program got the right solution or not. You can try as many times as you need.

In the actual contest you first need to solve the test phase before submitting the code, you must provide the source code used to solve the challenge and you can only submit once (once your solution is submitted you won't be able to amend it to fix issues or make it faster).

If you have any doubts, please check the [info section](#).

« Prev   Next »

Tweet about this! [#TuentiChallenge4](#)



Follow [@Tuentieng](#)