

একটি সমস্যা

$$(1 + 3 + 6 + 10 + \dots + S_n) \% 1000007 = ?$$

Handwritten notes for the first equation:

- $n=1$ points to 1
- $n=2$ points to 3, with $1+2$ written below it
- $n=3$ points to 6, with $1+2+3$ written below it
- $n=4$ points to 10, with $1+2+3+4$ written below it
- S_n has $1+2+\dots+n$ written below it
- At the top right, $n=4$ and $S_n \rightarrow S_4 \rightarrow 10$ are written.

$$S_n = 1 + 2 + 3 + \dots + n$$

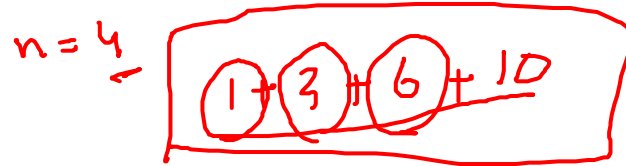
Handwritten calculation for S_4 :

$$S_4 = 1 + 2 + 3 + 4 = 10$$

এক সমস্যার দুই সমাধান – ভালো কে?

সমাধান - ০১

```
1 Long MOD = 1000007;  
2 Long solution01(Long n){  
3     Long sum = 0;  
4     Long item = 0;  
5     for (int i=1; i<=n; i++){  
6         item = (item + i) % MOD;  
7         sum = (sum + item) % MOD;  
8     }  
9     return sum;  
10  
11 }
```



সমাধান - ০২

```
1 Long MOD = 1000007;
2 Long solution02(Long n){
3     Long sum = 0;
4     for (int i=1; i<=n; i++){
5         Long item = 0;
6         for (int j=1; j<=i; j++){
7             item = (item + j) % MOD;
8         }
9         sum = (sum + item) % MOD;
10    }
11    return sum;
12 }
```

সমাধান - ০১ Vs সমাধান - ০২

```
1 Long MOD = 1000007;
2 Long solution01(Long n){
3     Long sum = 0;
4     Long item = 0;
5     for (int i=1; i<=n; i++){
6         item = (item + i) % MOD;
7         sum = (sum + item) % MOD;
8     }
9     return sum;
10
11 }
```

```
1 Long MOD = 1000007;
2 Long solution02(Long n){
3     Long sum = 0;
4     for (int i=1; i<=n; i++){
5         Long item = 0;
6         for (int j=1; j<=i; j++){
7             item = (item + j) % MOD;
8         }
9         sum = (sum + item) % MOD;
10    }
11    return sum;
12 }
```

সমাধান – ০১ Vs সমাধান – ০২

- একই environment (মেশিন, ল্যঙ্গুয়েজ)
- n এর বিভিন্ন মানের জন্য সমাধানের সময়
 - সমাধানের আগের সময় (start_time) 2.05
 - সমাধান চলবে (solution(n)) \rightarrow
 - সমাধানের পরের সময় (end_time) 2.10
 - সমাধানের সময় (time = end_time - start_time)
2.10 - 2.05

সমাধান - ০১ Vs সমাধান - ০২

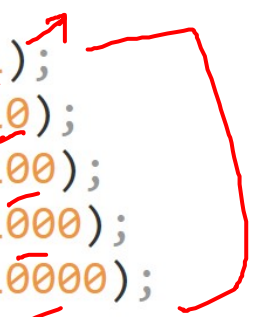
```
1 auto start_time = std::chrono::high_resolution_clock::now();  
2  
3 answer = solution(n);  
4  
5 auto end_time = std::chrono::high_resolution_clock::now();  
6 auto time = end_time - start_time;
```

সমাধান - ০১ Vs সমাধান - ০২

```
1 void calc(long n){
2     long answer;
3
4     cout << "For " << n << ":\n";
5
6     [auto start_time = std::chrono::high_resolution_clock::now();
7     answer = solution01(n);
8     auto end_time = std::chrono::high_resolution_clock::now();
9     auto time = end_time - start_time;
10
11     → cout << "Solve 01 -> " << answer << " " << std::chrono::duration_cast<std::chrono::mi
12
13     [start_time = std::chrono::high_resolution_clock::now();
14     answer = solution02(n);
15     end_time = std::chrono::high_resolution_clock::now();
16     time = end_time - start_time;
17
18     → cout << "Solve 02 -> " << answer << " " << std::chrono::duration_cast<std::chrono::mi
19 }
```


সমাধান – ০১ Vs সমাধান – ০২

```
1 int main(){  
2  
3     calc(1);  
4     calc(10);  
5     calc(100);  
6     calc(1000);  
7     calc(10000);  
8  
9     return 0;  
10 }
```

A red bracket is drawn on the right side of the code, spanning from line 3 to line 7. An arrow points from the top of the bracket to the argument '1' in the first call 'calc(1);'.

সমাধান - ০১ Vs সমাধান - ০২

For 1:

Solution 01 -> 1 0

Solution 02 -> 1 0

For 10:

Solution 01 -> 220 0

Solution 02 -> 220 1

For 100:

Solution 01 -> 171700 3

Solution 02 -> 171700 89

For 1000:

Solution 01 -> 165831 31

Solution 02 -> 165831 9178

For 10000:

Solution 01 -> 502995 417

Solution 02 -> 502995 869214

সমাধান - ০১

```
1 Long MOD = 1000007; → 1
2 Long solution01(Long n){ -
3     Long sum = 0; → 1,
4     Long item = 0; - 1,
5     for (int i=1; i<=n; i++){ -
6         item = (item + i) % MOD; -1] 2×n = 2^n
7         sum = (sum + item) % MOD; -1]
8     }
9     return sum; - 1,
10
11 }
```

$2n + 4$

সমাধান - ০২

```
1 Long MOD = 1000007; -1
2 Long solution02(Long n){-
3     Long sum = 0; -1
4     for (int i=1; i<=n; i++){-
5         Long item = 0; -1
6         for (int j=1; j<=i; j++){
7             item = (item + j) % MOD;
8         }
9         sum = (sum + item) % MOD; -1
10    }
11    return sum; -1
12 }
```

$$\underline{1+2+3 \dots n} = \left(\frac{n(n-1)}{2} + 2n \right) + 3$$

$$= \frac{n^2 - n}{2} + 2n + 3$$

$$= \frac{n^2 - n + 4n + 6}{2}$$

$$= \boxed{\frac{n^2 + 3n + 6}{2}}$$

সমাধান - ০১ Vs সমাধান - ০২

For 1:

Solution 01 -> 1 0

Solution 02 -> 1 0

For 10:

Solution 01 -> 220 0

Solution 02 -> 220 1

For 100:

Solution 01 -> 171700 3

Solution 02 -> 171700 89

For 1000:

Solution 01 -> 165831 31

Solution 02 -> 165831 9178

For 10000:

Solution 01 -> 502995 417

Solution 02 -> 502995 869214

$$n=100$$

$$S_1 \rightarrow 2n+4 = 2 \times 100 + 4 = 204$$

$$\begin{array}{r} 204 \text{ --- } 3 \\ 1 \text{ --- } 204 \end{array} \quad \text{--- } \frac{3}{204} \quad \text{--- } 1015$$

$$\begin{aligned} S_2 \rightarrow \frac{n^2+3n+6}{2} &= \frac{100^2+3 \times 100+6}{2} \\ &= \frac{10000+300+6}{2} \\ &= \frac{10306}{2} \\ &= 5153 \end{aligned}$$

$$\begin{array}{r} 5153 \text{ --- } 89 \\ 1 \text{ --- } 89 \\ \text{--- } 5153 \end{array} \quad \text{--- } 017$$

সমাধান - ০১

```
1 void calc(long n){
2     long answer;
3
4     cout << "\nFor " << n << ":\n";
5
6     auto start_time = std::chrono::high_resolution_clock::now();
7
8     answer = solution01(n);
9     auto end_time = std::chrono::high_resolution_clock::now();
10    auto time = end_time - start_time;
11    cout << "Solution 01 -> " << answer << " " << std::chrono::duration_cast<std::chronoc
12
13
14    start_time = std::chrono::high_resolution_clock::now();
15    answer = solution01(n);
16    end_time = std::chrono::high_resolution_clock::now();
17    time = end_time - start_time;
18
19    cout << "Solution 01 (Re) -> " << answer << " " << std::chrono::duration_cast<std:::c
20 }
```

সমাধান – ০১

```
1  int main(){  
2  
3      calc(1);  
4      calc(10);  
5      calc(100);  
6      calc(1000);  
7      calc(10000);  
8      calc(100000);  
9  
10     return 0;  
11 }
```

সমাধান - ০১

For 10:

Solution 01 -> 220 0

Solution 01 (Re) -> 220 0

For 100:

Solution 01 -> 171700 3

Solution 01 (Re) -> 171700 3

For 1000:

Solution 01 -> 165831 34

Solution 01 (Re) -> 165831 34

For 10000:

Solution 01 -> 502995 347

Solution 01 (Re) -> 502995 334

For 100000:

Solution 01 -> 6500 4542

Solution 01 (Re) -> 6500 3404

```
1 Long MOD = 1000007;
2 Long solution01(Long n){
3     Long sum = 0;
4     Long item = 0;
5     for (int i=1; i<=n; i++){
6         item = (item + i) % MOD;
7         sum = (sum + item) % MOD;
8     }
9     return sum;
10
11 }
```

$2n+4$

সমাধান – ০১ Vs সমাধান – ০২

For 1:

Solution 01 -> 1 0

Solution 02 -> 1 0

For 10:

Solution 01 -> 220 0

Solution 02 -> 220 1

For 100:

Solution 01 -> 171700 3

Solution 02 -> 171700 89

For 1000:

Solution 01 -> 165831 31

Solution 02 -> 165831 9178

For 10000:

Solution 01 -> 502995 417

Solution 02 -> 502995 869214

Super AI ML Media Player

- 144px – 1080px
- 15fps – 60fps
- 1s -> 2s
- it is as good as dead.

