Lab Sheet - 3

The purpose of this lab is to teach you how to **break down a problem** into a series of **smaller to increasingly bigger problems** so that they could be solved in a **progressive** way. Essentially you will be able to conquer the solution in a stepwise manner. 4 problems in Codeforces are considered. By the time you start this lab, your instructor would have already introduced these problems and discussed the stepwise breaking down. You can first solve them in Colab in a progressive fashion, try out with different inputs, ensure the program works as per expectation and then submit in Codeforces.

- First read the problem and make a judgement if you can solve it directly. If not, break down the problem into a **series of stages** so that you can solve them one at a time.
- Think how you can **solve each stage** by making **delta improvements** to the previous stage. Note, each subproblem has its own objective. Test cases must be adapted.
- **Solve each stage completely** and move to next stage. i.e Solve stage 1, then proceed to stage 2, then stage 3, so on and so forth.
- Refer to Python Basics for syntax. It should be sufficient to code.

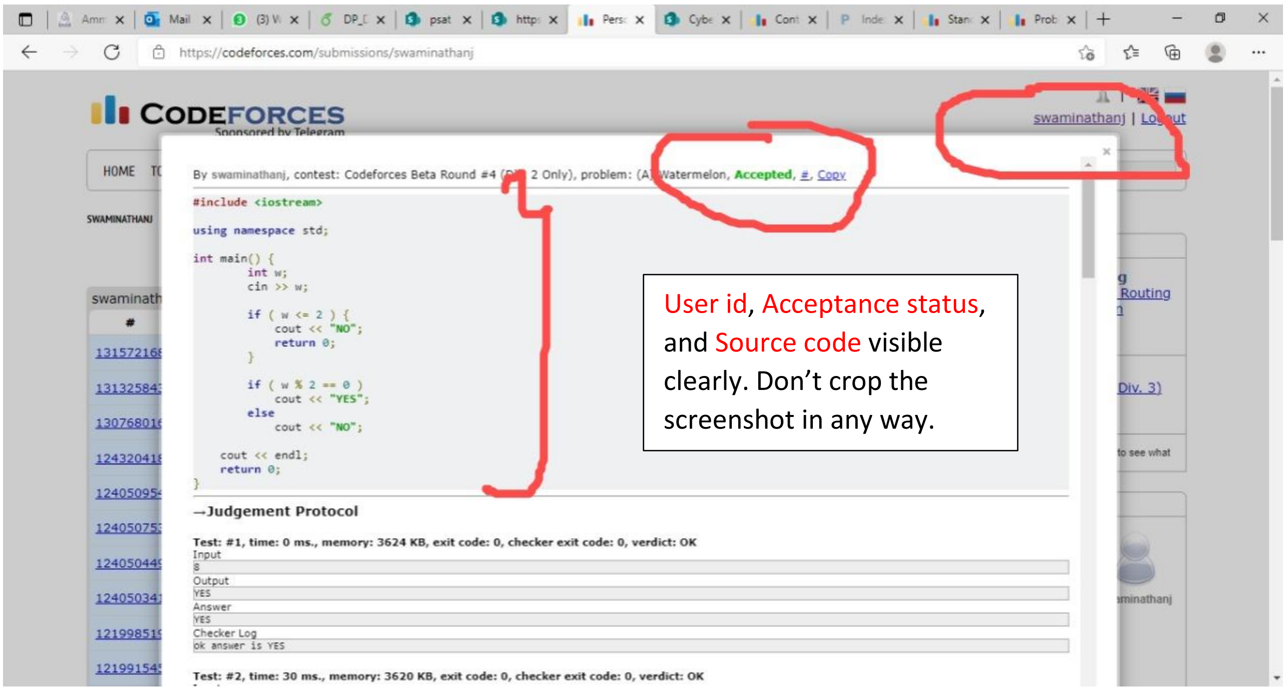
Note: The **way** you break down **depends on** the **way you view** the problem. The **number of stages** depends on the problem and your **confidence level**.

1. Frog Jumping (1077A)
   a. https://codeforces.com/problemset/problem/1077/A
   b. A simple math-based problem. Even / odd case to be dealt with differently.
   c. You need to use the right operators in a right way to get the right result.
2. The Rank (1017A)
   a. https://codeforces.com/problemset/problem/1017/A
   b. You need to compute intermediate result and use it for finding final result.
   c. You need a loop to repeat finding sum and compare with intermediate.
3. Golden Plate (1031A)
   a. https://codeforces.com/problemset/problem/1031/A
   b. A simple math-based problem. Computation based on width and height.
   c. Compute for one ring, fixed number of rings and any number of rings.
4. Bear and Big Brother (791A)
   a. https://codeforces.com/contest/791/problem/A.
   b. Yet another math-based problem.
   c. You need to loop until a certain criterion is met (not fixed no. of times). You will be taught how to do conditional looping using **while** construct.

Note: Please ensure the solutions are submitted and accepted in Codeforces.

Important note: You must try on your own and drill your way through. Don't mind if you struggle. Without struggle, there is no learning. Get used to the struggling and overcoming. That is very normal in problem solving. You can seek help after you have tried enough.

Submission: The proof of acceptance (screenshots) along with source code for all 4 problems must be submitted in Teams → Week 3 - Assignment. Sample format below.



By swaminathanj, contest: Codeforces Beta Round #4 (Div. 2 Only), problem: (A) Watermelon, Accepted, #, Copy

```
#include <iostream>

using namespace std;

int main() {
        int w;
        cin >> w;

        if ( w <= 2 ) {
            cout << "NO";
            return 0;
        }

        if ( w % 2 == 0 )
            cout << "YES";
        else
            cout << "NO";

    cout << endl;
    return 0;
}
```

→Judgement Protocol

```
Test: #1, time: 0 ms., memory: 3624 KB, exit code: 0, checker exit code: 0, verdict: OK
Input
8
Output
YES
Answer
YES
Checker Log
ok answer is YES

Test: #2, time: 30 ms., memory: 3620 KB, exit code: 0, checker exit code: 0, verdict: OK
```

User id, Acceptance status, and Source code visible clearly. Don’t crop the screenshot in any way.

The code shown above is in C++. You submission would have Python code instead.