Warming Up With Competitive Programming

1) Problem Code :- **INTEST**

Problem Name :- **Enormous Input Test**

```cpp
#include <bits/stdc++.h>
using namespace std;

int main() {
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);

    int n, k;
    cin >> n >> k;
    int ans = 0;

    for (int i = 0; i < n; i++) {
        int t;
        cin >> t;
        if (t % k == 0) {
            ans++;
        }
    }
    // Print the ans.
    cout << ans << "\n";

    return 0;
}
```

2) Problem Code :- **HS08TEST**

Problem Name :- **ATM**

```cpp
#include <iostream>
using namespace std;

int main()
{
    // your code goes here
    int widraw;
    float balance;
    cin >> widraw;
    cin >> balance;
    if (widraw % 5 == 0 && (widraw + 0.5) < balance)
    {
        balance = balance - (float)widraw - 0.5;
        cout << balance;
    }
    else
    {
        cout << balance;
    }

    return 0;
}
```

3) Problem Code :- **FLOW017**

Problem Name :- **Second Largest**

```cpp
#include <iostream>
using namespace std;
int main()
{
    long int T, A, B, C;
    cin >> T;
    for (int i = 1; i <= T; i++)
    {   cin >> A >> B >> C;
        if (A > B && A > C)
        {   if (B > C)
                {cout << B << "\n";}
            else
                {cout << C << "\n";}
        }
        else{
            if (B > C){
                if (C > A)
                {cout << C << "\n";}
                else
                {cout << A << "\n";}
            }
            else{
                if (B > A)
                {cout << B << "\n";}
                else
                {cout << A << "\n";}
            }}}
    return 0;
}
```

4) Problem Code :- **FLOW011**

Problem Name :- **Gross Salary**

```cpp
#include <iostream>
#include<iomanip>
using namespace std;
int main()
{
    int T;
    cin >> T;
    while (T--)
    {
        int salary;
        cin >> salary;
        float gross_salary;
        if (salary < 1500)
        {
            gross_salary = salary * 2;
            cout << gross_salary << "\n";
        }
        else
        {
            gross_salary = (salary * 1.98) + 500;
            cout <<fixed<< setprecision(2) << gross_salary << "\n";
        }
    }
}
```