

Number Logic Problems

These are very common in entry-level coding rounds.

Problem 1: Prime number

C++ solution

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

bool isPrime(int n) {
    if (n < 2) return false;
    if (n == 2) return true;
    if (n % 2 == 0) return false;
    for (int i = 3; i * i <= n; i += 2) {
        if (n % i == 0) return false;
    }
    return true;
}

int main() {
    int n;
    cin >> n;
    cout << (isPrime(n) ? "YES" : "NO");
    return 0;
}
```

Problem 2: Factorial

C++ solution

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

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

    long long fact = 1;
    for (int i = 1; i <= n; i++) {
        fact *= i;
    }

    cout << fact;
}
```

```
    return 0;
}
```

Problem 3: Fibonacci series

Print first n Fibonacci numbers.

C++ solution

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

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

    long long a = 0, b = 1;
    for (int i = 0; i < n; i++) {
        cout << a;
        if (i != n - 1) cout << " ";
        long long c = a + b;
        a = b;
        b = c;
    }
    return 0;
}
```

Problem 4: GCD

C++ solution

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

int main() {
    int a, b;
    cin >> a >> b;

    while (b != 0) {
        int r = a % b;
        a = b;
        b = r;
    }

    cout << a;
```

```
    return 0;
}
```

Problem 5: Reverse number

C++ solution

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

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

    int rev = 0;
    while (n > 0) {
        int d = n % 10;
        rev = rev * 10 + d;
        n /= 10;
    }

    cout << rev;
    return 0;
}
```

Problem 6: Armstrong number

An Armstrong number is equal to the sum of its digits each raised to the number of digits.

Example:

```
153 = 1^3 + 5^3 + 3^3
```

C++ solution

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

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

    int temp = n;
    int digits = 0;
    while (temp > 0) {
        digits++;
    }
}
```

```

        temp /= 10;
    }

    temp = n;
    int sum = 0;
    while (temp > 0) {
        int d = temp % 10;
        sum += pow(d, digits);
        temp /= 10;
    }

    cout << (sum == n ? "YES" : "NO");
    return 0;
}

```

Problem 7: Sum of digits

C++ solution

```

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

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

    int sum = 0;
    while (n > 0) {
        sum += n % 10;
        n /= 10;
    }

    cout << sum;
    return 0;
}

```