Lab 1 Isa Dzhumabaev.

Program is a solution for online judge problem:

https://www.urionlinejudge.com.br/judge/en/problems/view/1568

Solution is based on prime numbers and made using Eratosthenes sieve. As first step algorithm finds all prime numbers from 0 to 1 000 000, then program factorizes the input number and counts quantity of its factors, this quantity will be the answer.

Source code:

```
#include "stdio.h"
#include "inttypes.h"
#define N 1000000
int findPrimes(int*);
int main(int argc, char** argv)
   int primes[N];
    int p = findPrimes(primes);
    int64 t n;
    while(scanf("%lld", &n) != EOF) {
        if(n == 0) {
           printf("1\n");
            continue;
        while(n % 2 == 0) {
           n /= 2;
        int r = 1;
        int i = 0;
        while ( i < p && n != 1) {
            int count = 0;
            while(n % primes[i] == 0) {
               n /= primes[i];
               count++;
            r *= count + 1;
            i++;
        r *= n != 1 ? 2 : 1;
        printf("%d\n", r);
```

return 0;

```
}
int findPrimes(int* primes) {
    char s[N] = \{ 0 \};
    primes[0] = 2;
    int p = 1;
    int i = 2;
    while (i < N)
        if(s[i] == 0) {
            primes[p++] = i;
            int j = i;
            while (j < N) {
                s[j] = 1;
                j += i+i;
        }
        i++;
    return p;
}
```

Program was tested using Perf profiling.

Result:

```
Performance counter stats for './Isa':
                                                      0,984 CPUs utilized
       402,168465
                       task-clock (msec)
                                                 #
            1 237
                       context-switches
                                                      0,003 M/sec
                                                 #
                0
                       cpu-migrations
                                                 #
                                                      0,000 K/sec
                                                      0,922 K/sec
              371
                       page-faults
  <not supported>
                       cycles
                       instructions
  <not supported>
  <not supported>
                       branches
  <not supported>
                       branch-misses
  <not supported>
                       L1-dcache-loads
                       L1-dcache-load-misses
  <not supported>
  <not supported>
                       LLC-loads
```

Input:

```
8999964000011
899998380000053
899998200000059
899997780000073
899997660000077
```

- -----

- 0,7,70,7000003003

- -----

899987160038227

899986980038989