Given an array containing n integers. Sort it.

The only line of the input contains two integers n and a_0 – the number of elements in the array and the value of the element with index 0 ($1 \le n \le 5 \cdot 10^7$; $0 \le a_0 < 2^{20}$).

The remaining elements of the array must be generated. Array elements are set using a pseudo-random generator according to the formula: $a_i = (1103515245 a_{i-1} + 12345) \bmod 2^{20}$. To fill array elements with initial values, you can use the following C++ function:

```
void fill(vector<int>& a, int n, int start)
{
    a.resize(n);
    a[0] = start;
    for (int i = 1; i < n; ++i) {
        a[i] = (1103515245LL * a[i - 1] + 12345) % (1 << 20);
    }
}
```

The program should sort this array and output the value of the hash function from it. To calculate this value, it is necessary to sequentially, starting from the value 0, multiply this value by 239, add the next element of the sorted array and take its remainder after division by 1 000 000 007. This value can be calculated using the following C++ function:

```
int calc_hash(const vector<int>& a)
{
    int answer = 0;
    for (int i : a) {
        answer = (239LL * answer + i) % 1000000007;
    }
    return answer;
}
```

Sample input:

2 2
0 5
7 10
1 6

Sample output:

1 0