N - Bachgold Problem

Bachgold problem is very easy to formulate. Given a positive integer *n* represent it as a sum of **maximum possible** number of prime numbers. One can prove that such representation exists for any integer greater than 1.

Recall that integer *k* is called prime if it is greater than 1 and has exactly two positive integer divisors — 1 and *k*.

**Input**

The only line of the input contains a single integer *n* (2 ≤ *n* ≤ 100 000).

**Output**

The first line of the output contains a single integer *k* — maximum possible number of primes in representation.

The second line should contain *k* primes with their sum equal to *n*. You can print them in any order. If there are several optimal solution, print any of them.

**Examples**

**Input**

5

**Output**

2  
2 3

**Input**

6

**Output**

3  
2 2 2

#include <iostream>

using namespace *std*;

int main()

{

int n; *cin*>> n;

auto num2 = 2;

auto num3 = 3;

*cout* << n / 2 << *endl*;

if (n % 2 == 0)

{

for (auto i = 0; i < n / 2; i++)

*cout* << 2 << " ";

*cout* << *endl*;

}

else

{

for (auto i = 0; i < (n - 2) / 2; i++)

*cout* << 2 << " ";

*cout* << 3 << *endl*;

}

return 0;

}