H - k-Factorization

 Given a positive integer *n*, find *k* integers (not necessary distinct) such that all these integers are strictly greater than 1, and their product is equal to *n*.

**Input**

The first line contains two integers *n* and *k* (2 ≤ *n* ≤ 100000, 1 ≤ *k* ≤ 20).

**Output**

If it's impossible to find the representation of *n* as a product of *k* numbers, print-1.

Otherwise, print *k* integers in any order. Their product must be equal to *n*. If there are multiple answers, print any of them.

**Example**

**Input**

100000 2

**Output**

2 50000

**Input**

100000 20

**Output**

-1

**Input**

1024 5

**Output**

2 64 2 2 2

#include<iostream>

#include<stdio.h>

#include<cstring>

#include<queue>

using namespace std;

queue<int> q;

int main()

{

int n,k;

cin>>n>>k;

if(k==1)

{

cout<<n;

return 0;

}

for(int i=2;n!=1;i++)

{

if(n%i==0)

{

n/=i;

q.push(i);

if(q.size()==(k-1)&&n!=1)

{

q.push(n);

while(q.size()!=0)

{

cout<<q.front()<<' ';

q.pop();

}

return 0;

}

i--;

}

}

cout<<"-1";

return 0;

}