Problem C. Minimum divisor of a number

A natural number n>1 is given. Print its smallest divisor other than 1.

Form the solution in the form of the MinDivisor (n) function. The algorithm must be O (n).
Indication. If the number n has no divisor not exceeding n, then the number n is prime and the answer will be the number n itself.

Input data
A natural number is entered.

Output
Print the answer to the problem.

Examples of
input data
4

output
2

input data
5

output
5

```
int N;

// N = 9

// N

std::cin >> N;

  for (int i = 2; i <= N; i++ ) {

     if (N % i == 0 ) {

        std::cout << " The least divisor is " << i << std::endl;

        break;

     }
```

12
6

17
not exist

Maximum divisor of a number not equal to a number or print it not exist

```
int N;
int maxDivisor;
int status = 0; // Status has not encountered divisor

// N = 12
// max divisor = 6
std::cin >> N;
  for (int i = 2; i < N; i++ ) {
     if (N % i == 0 ) {
        maxDivisor = i;
        status = 1;
     }
  }
  if ( status == 0) {
     std::cout << " Does not exist " << std::endl;
  }
  else {
     std::cout << " Max divisor is " << maxDivisor << std::endl;
  }
```

detect prime or not prime number

```
for (int i = 2; i <= N; i++ ) {
   if (N % i == 0 ) {
      if ( i == N ) {
         std::cout << " Prime " << std::endl;
         break;
      }
      else if ( i != N) {
         std::cout << " Not Prime." << std::endl;
         break;
      }
   }
}
```