A number is called *palindrome\_prime* if it is palindromic and it is prime. We assume a number is palindromic if it reads the same backward or forward.  
Here is the problem: Given a positive integer N, your job is to find out whether it is a *palindrome\_prime*. If it is a *palindrome\_prime*, please return true. Otherwise please return false.

Note:  
1 is not a prime.

* **[input] integer N**
  + A positive number N (1 ≤ N ≤ 1,000,000).
* **[output] boolean**
  + Please output the answer.

<https://codefights.com/challenge/bSv8HvMCcNdfgQgLo>

--ACEPTADO--

bool palindrome\_prime(int N) {

//es palindromo

int rev = 0;

int copia = N;

while(copia > 0){

rev = rev \* 10 + (copia % 10);

copia /=10;

}

if(rev == N) { //es palindromo

//me fijo si es primo

if(N < 2)return false;

if(N == 2)return true;

if(N % 2 == 0)return false;

int sqr = sqrt(N);

for(int i = 3; i <=sqr; i+=2) {

if(N % i==0) return false;

}

}else{

return false;

}

return true;

}