题目：

Description:

Count the number of prime numbers less than a non-negative number, *n*.

1.时间：O（ ）; 空间：O（1） ->超时

```
class Solution {
public:
    int countPrimes(int n) {
        int result = 0;
        for (int i = 2; i < n; ++i){
            if (isPrime(i)){
                result++;
            }
        }
        return result;
    }
private:
    bool isPrime(int num){
        if (num == 2 || num == 3) return true;
        const int Upper = std::sqrt(num);
        for (int i = 2; i <= Upper; ++i){
            if (num % i == 0) return false;
```

```
        }
        return true;
    }
};
```

2.时间：O（ ）; 空间：O（N）　　->哈希加速

```
class Solution {
public:
    int countPrimes(int n) {
        int result = 0;
        std::vector<int> searchTable(n, 0);
        for (int i = 2; i < n; ++i){
            if (searchTable[i] == 0 && isPrime(i)){
                result++;
                for (int k = i; k < n; k += i){
                    searchTable[k] = 1;
                }
            }
        }
        return result;
    }
private:
```

```
    bool isPrime(int num){
        if (num == 2 || num == 3) return true;
        const int Upper = std::sqrt(num);
        for (int i = 2; i <= Upper; ++i){
            if (num % i == 0) return false;
        }
        return true;
    }
};
```

3. 时间：O（ ）；空间：O（N） ->哈希加速

```
/* 每个合数必有一个最小素因子，根据每个最小因子去访问合数就能防止合数被重复访问
*/
class Solution {
public:
    int countPrimes(int n) {
        std::vector<int> searchTable(n, 0);     /* 1 表示肯定不是素数 */
        std::vector<int> primeNum;  /* 记录素数 */
        for (int i = 2; i < n; ++i){
            if (searchTable[i] == 0){
                primeNum.push_back(i);
            }
```

```
            for (int k = 0; k < primeNum.size() && (primeNum[k] * i < n); ++k){
                searchTable[primeNum[k] * i] = 1;
                if (i % primeNum[k] == 0) break;
            }
        }
        return primeNum.size();
    }
private:
    bool isPrime(int num){
        if (num == 2 || num == 3) return true;
        const int Upper = std::sqrt(num);
        for (int i = 2; i <= Upper; ++i){
            if (num % i == 0) return false;
        }
        return true;
    }
};
```