

截圖

Source Code

```
1  int countPrimes(int n){
2      int cnt=0,*table=calloc(n,sizeof(int));
3      for(int i=2;i<n;i++){
4          if(table[i]==0){
5              cnt++;
6              for(int j=1;j*i<n;j++){
7                  table[i*j]=1;
8              }
9          }
10     }
11     return cnt;
12 }
```

解釋

給定一個數字n，找出小於n的所有質數，用的是隱藏提示中的Sieve of Eratosthenes方法，該方法是找到一個質數，把質數的倍數標記起來，因為質數的倍數不為質數，把這些排除後就可以找到我們需要的所有質數了。

一開始先開一個計數器，跟一個陣列(Line2)，查詢一個數是否為質數，接著遍歷(Line3)，1不為質數，所以從2開始，一直找到n，如果table[i]=0，也就是說i為質數(Line4)，就讓計數器+1(Line5)，並且讓他標記所有i的整數倍數為1(Line6~8)，最後回傳得到的計數器數值就好(Line11)。