

Sieves of Erasthóthenes

iaacornus

August 31, 2022

Prime Numbers

Given a number N , find all the prime numbers smaller than N , where N is an integer $< 10^6$.

Solution

Using sieves of erasthóthenes, the prime numbers smaller than N can be found:

```
1 def sieves_of_erasthóthenes(N: int) -> list[int]:
2     """Finds the prime numbers from range N
3
4     Args:
5         N -- the upper limit
6
7     Returns:
8         The list of primes smaller than N.
9     """
10
11     not_primes: list[int] = [True]*N
12     for i in range(2, N):
13         if not_primes[i]:
14             for j in range(i, N, i):
15                 not_primes[j] = False
16
17     return [x for x in range(N) if not not_primes[x]]
```

Sieves of erasthóthenes can also be implemented in C:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 #define ARR_LIMIT 10000000
5
6 int main(){
7     unsigned long long int i,j;
8     int *primes;
9     int z = 1;
10
11     primes = malloc(sizeof(int)*LIMIT);
12
13     for (i=2;i<limit;i++) {
14         primes[i]=1;
15     }
16
17     for (i=2;i<limit;i++) {
18         if (primes[i]) {
19             for (j=i*i;j<limit;j++) {
20                 primes[i*j]=0;
21             }
22         }
23     }
24
25     for (i=2;i<limit;i++) {
26         if (primes[i]) {
27             printf("%dth prime = %dn",z++,i);
```

```
28     }
29 }
30
31 return 0;
32 }
```