Compilando Conocimiento

Refence

Competitive Programming

Rosas Hernandez Oscar Andrés

July 2018

Contents

Part I

Things to Learn / To Do

Chapter 1

C++

1.1 Sieve of Eratosthenes

```
#include <cstdint>
int8_t likeChar {};
int16_t likeShort {};
int32_t likeInt {};
int64_t likeLong {};

// And the unsigned versions :
uint8_t likeChar {};
uint16_t likeShort {};
uint32_t likeInt {};
uint64_t likeLong {};
```

Part II

Number Theory

Chapter 2

Primes

2.1 Sieve of Eratosthenes

2.1.1 Get the Boolean Version

```
    std::vector<bool> isPrime(n + 1, true);
      //Ok, first, allocate space
    isPrime[0] = isPrime[1] = false;
      //Now, 0 & 1(maybe) are not prime

    for (T i = 4; i <= n; i += 2) isPrime[i] = false;
      //Eliminate all the evens numbers

    for (T i = 3; i * i <= n; i += 2)
      //For every odd number < n
        if (isPrime[i])
      //If we found a prime :0
            for (T j = i * i; j <= n; j += 2 * i)
      //ForEach multiple we have'nt check
              isPrime[j] = false;
      //Each multiple is not prime

    return isPrime;
      //Return the complete sieve
}

// *******   ERATOSTHENES SIEVE / VECTOR OF PRIMES   *******
```

2.1.2 Get the Vector of Primes

```
std::vector<T> EratosthenesSievePrimes(T n) {
      //Return a vector of only primes
    std::vector<bool> isPrime(n + 1, true);
      //Create the origianl Sieve
    std::vector<T> Primes{2};
      //2 is a prime, dahhhhh!

    //isPrime[0] = isPrime[1] = false;
      //Uncomment if you want bool version
    //for (T i = 4; i <= n; i += 2) isPrime[i] = false;
      //Uncomment if you want bool version

    for (T i = 3; i <= n; i += 2) {
      //For every odd number < n
        if (isPrime[i]) {
      //If we still believe it's a prime
            Primes.push_back(i);
      //Add it to the vector, it's a prime

            if (i * i <= n)
      //It make sense to delete multiples?
              for (T j = i * i; j <= n; j += 2 * i)
      //ForEach multiple we have'nt check
                isPrime[j] = false;
      //Each multiple is not prime
        }
    }

    return Primes;
      //Return the vector of only primes
}
```