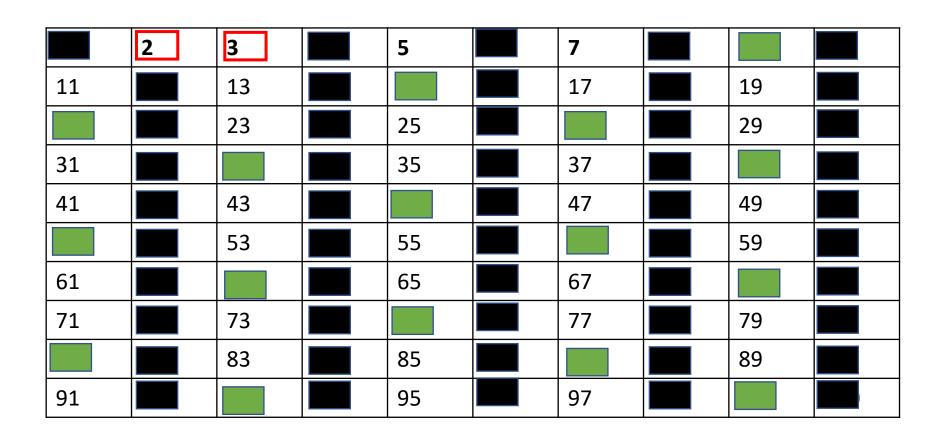
Sieve of Eratosthenes

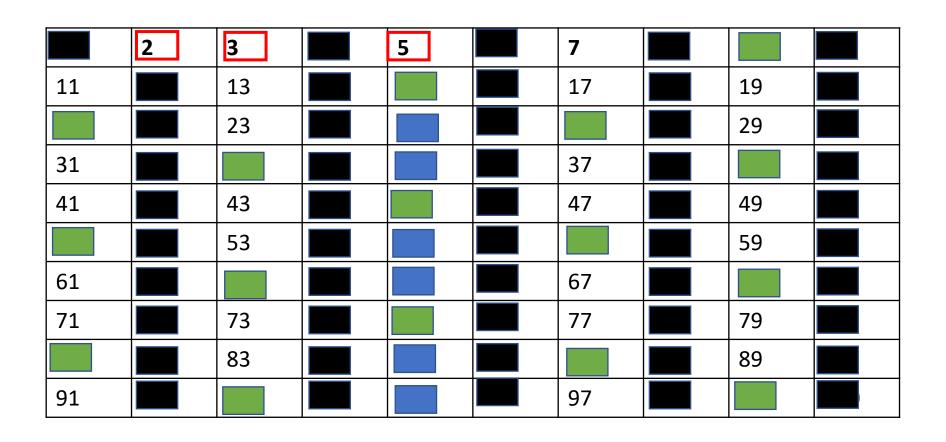
Algorithm

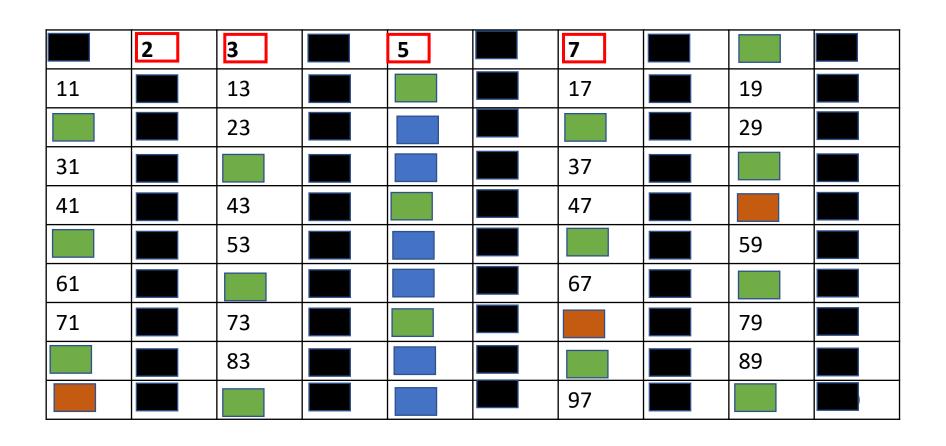
- Start by assuming a list of all numbers starting from 2 are all prime until proven otherwise
 - Loop through all numbers not yet proven to be not prime, and assume all multiples of that number are not prime

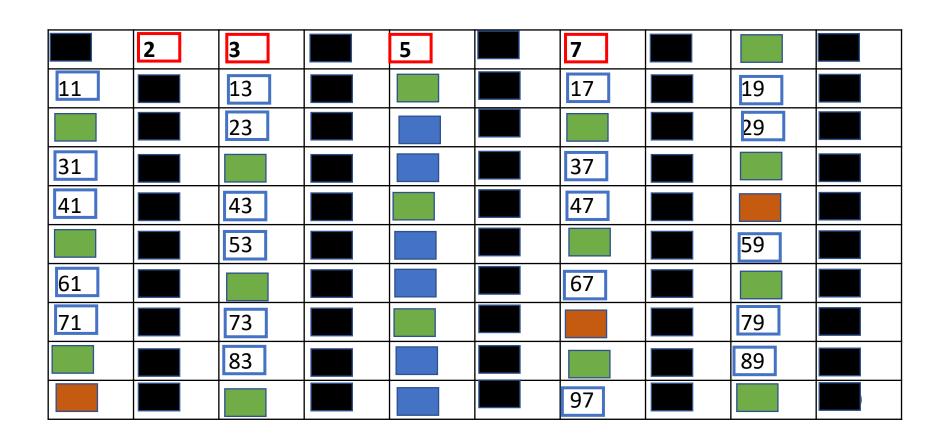
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

	2	3	5	7	9	
11		13	15	17	19	
21		23	25	27	29	
31		33	35	37	39	
41		43	45	47	49	
51		53	55	57	59	
61		63	65	67	69	
71		73	75	77	79	
81		83	85	87	89	
91		93	95	97	99	









Loop Carried Dependencies

• Define where there is and is not loop carried dependency in the following nested loop. How does this affect parallelization?

```
FOR I = 2:SQRT(N)
IF (I IS PRIME)
    FOR J = I*I:N
    J IS NOT PRIME
```