

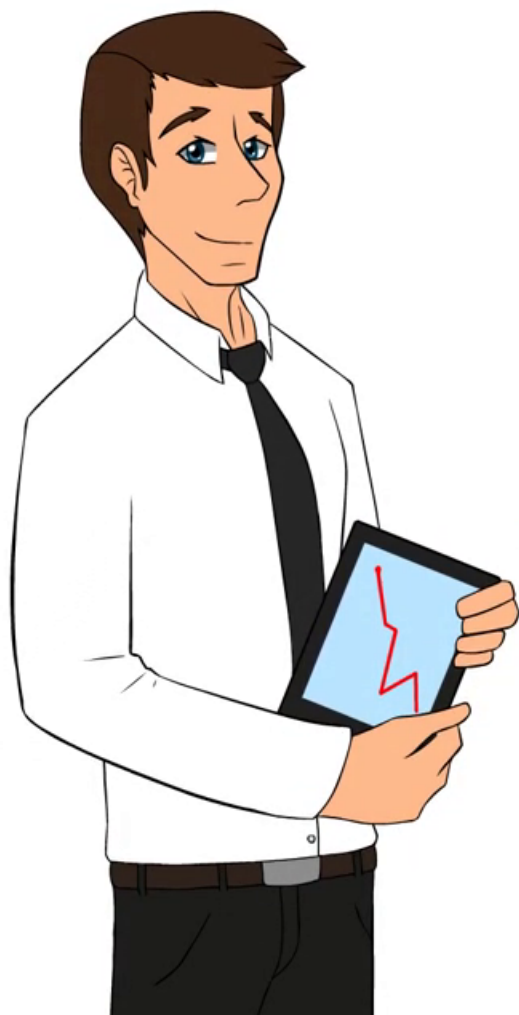


welcome
linear sieve



prior knowledge

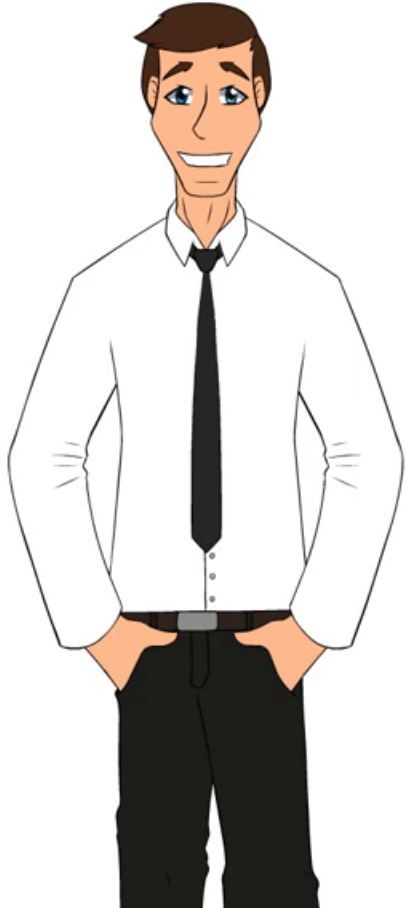
sieve of eratosthenes



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

Numbers that divide by 2 in GREEN
Numbers that divide by 3 in BLUE
Numbers that divide by 5 in ORANGE
Numbers that divide by 7 in PURPLE

optimizations



```
bool composite[sz];
vector<int>prime;
void linear_sieve (int n)
{
    composite[0]=composite[1]=1;
    for (int i = 2; i<n; ++i)
    {
        if (!composite[i]) prime.push_back (i);
        for (int j = 0; j < prime.size () && i * prime[j] < n; ++j)
        {
            composite[i * prime[j]] = true;
            if (i % prime[j] == 0) break;
        }
    }
}
```

for(i,n)	if (prime[j]*i<n) for (j,prime vector)	2 3 4 5 6 7 8 9 10 11
i	prime vector	composite[i*prime[j]]=1
2	2	4
3	2 3	6 9
4	2 3	8 10
5	2 3	



what did you learn at this video ?

**linear sieve with
time complexity $O(n)$**

