Algorithm :

```
sum = 0;
FOR (i = 1; i <= 100; i++) {
   prime_Num = true;                  // Assume the number is prime
   IF (i == 1) {
      prime_Num = false;              // 1 is not a prime number
   }
   FOR (j = 2; j <= sqrt(i); j++) {   // Loop through numbers from 2 to the square root of i
      IF (i % j == 0) {
         prime_Num = false;           // If divisible, i is not a prime number
         BREAK;
      }
   }
   IF (prime_Num == true) {
      sum = sum + i;                  // Add i to the sum
      print i, sum;
   }
}
print sum;
```

Analysis :

Variable: sum, i, j, prime_Num
Condition: i <= 1000, i==1, j<sqrt(i), i %j ==0, prime_Num==true
Output: i, sum

Step/Line no	i	i <= 100	i == 1	j	j <= sqrt(i)	i % j == 0	prime_Num	sum	Output
1	-	-	-	-	-	-	-	0	
2	1	true	true	-	-	-	false	0	
3	2	true	false	-	-	-	true	2	2, 2
4	3	true	false	2	false	false	true	5	3, 5
5	4	true	false	2	true	true	false	5	

6	5	true	false	2	true	false	true	10	5, 10
7	6	true	false	2	true	true	false	10	
8	7	true	false	2	true	false	true	17	7, 17
9	8	true	false	2	true	true	false	17	
10	9	true	false	2	true	true	false	17	
11	10	true	false	2	true	true	false	17	
12	11	true	false	2	true	false	true	28	11, 28
13	12	true	false	2	true	true	false	28	
14	13	true	false	2	true	false	true	41	13, 41
…	…	…	…	…	…	…	…	…	…
n	97	true	false	2	true	false	true	1060	97, 1060
n+1	98	true	false	2	true	true	false	1060	
n+2	99	true	false	2	true	true	false	1060	
n+3	100	true	false	2	true	true	false	1060	
n+4	101	false	-	-	-	-	-	1060	