CS569 Assignment 1 Write-up

Yuan Zhang

April 21, 2015

What I found, the primes.c program, for this assignment was actually correct. However, it's hard to put the code on CBMC platform to check the correctness. The part that is difficult to specify is there were three funcions that were called ex, primes, and markCom, respectively. Moreover, there were some nested loop right in the three funcions, which means when I deal with the prime function it had to call the ex and markCom funcions to generate another two loops. I wrote the harness c file like this:

```
int main(int argc, char const *argv[])
{
    ...
    int i, n, j;
    int pnum = 0;
    unsigned int s = nondet_unsigned_int();
    __CPROVER_assume (s <= SIZE);
    n = primes(a, SIZE);
    for(i = 0; i < n; i ++) {
        for(j = 3; j < a[n]; j ++) {
            if (a[i] % j != 0) {
                pnum += 1;
            }
        }
    }
    assert (pnum == n);
    return 0;
}
```

I was tring to use a double loop to check if all the numbers generated by original prime.c program are exactly primes by comparing the amount of numbers from each other.

Saying about the result, no matter what value I assigned to SIZE and loop limit, the same error, unwinding assertion loop 0, always came up. If I turned off the bound of loop, I got a horrible runtime, almost like running infinitly. According to what I learned about CBMC so far, it is a preliminary static analysis tool, based on computing a fixed point on various abstract domains. For some reasons, CBMC is an useful static analysis, such as for loop unwinding. Because CBMC checks bounded model, all loops should have a finite runtime in order to make sure all bugs could be found. However, it might be wrong about what I did, but the loop checking is not working well for a nested loop, and I don't know what happend here.