Cryptography

Diego Alfonso Prieto Torres - Sebastian Camilo Martinez Reyes

25 de octubre de 2012

Índice

1. Contextualizacion

El problema de Cryptography es un problema usado en maratones de programacion cuyo enunciado puede encontrarse actualmente en el Juez en Linea TIMUS identificado con el codigo 1086. Este es un problema que esta clasificado dentro de la teoria de numeros, y por lo tanto, es la oportunidad perfecta para desafiar los conocimientos matematicos, combinados con las habilidades en programacion. El documento describe la estrategia que se uso para dar una solucion eficiente y asi el estudiante se desarrolle en los contextos de programacion y matematicos de forma fluida.

2. Definicion del Problema

2.1. Objetivos

- Encontrar el $i\ th$ primo, es el objetivo principal del programa.
- Lograr combinar las teorias matematicas junto con las habilidades en programacion, para lograr escribir una solucion eficiente.

2.2. Precondicion

Como precondicion tenemos un numero n que representa el $i\ th$ primo a hallar y $1 \leq n \leq 15000$.

2.3. Poscondicion

Lo que debemos entregar es el $i\ th$ primo.

3. Definicion del Problema

Queremos escribir un programa que encuentre el $i\ th$ numero primo ordenado.

3.1. Definicion de Conceptos

Se define numero primo como el numero tal que no es divisible mas que por el mismo y por la unidad. La sucesion infinita f, con elementos pertenecientes al conjunto S, tal que S es el conjunto de numeros primos, se efine como:

$$S = \left\{ i \mid 2 \leq i \wedge \neg(\exists j \mid 2 \leq j \leq \sqrt{i} : i \mathbf{mod} j = 0) : i \right\}$$

$$f : \{1, 2, ..., \infty\} \longrightarrow S$$

Luego tenemos:
$f(2) = 2$
$f(3) = 5$
$f(5) = 11$
$f(7) = 17$

3.2. Introduccion al Problema

Como el problema consiste en informar el $i\ th$ primo, basta con informar el valor de la sucesion en ese estado, es decir $f(n)$ es el valor que estamos buscando.

3.3. Estrategia de la Solucion

La solucion consiste en encontrar el conjunto de los numero primos y guardarlos en una sucesion de manera que podamos informar rapidamente el resultado; pero, debido a que computacionalmente es mas muy costoso hacer este proceso, es mas sencillo, de cierto modo, encontrar el conjunto de los valores que no son primos y hallar a continuacion su complemento.

Hecho esto, ya solo basta con informar cual es el valor de la sucecion en la $i\ th$ posicion, de esta manera cumplimos con los objetivos del problema.

3.4. Leve Nocion de Estructura de Datos

Como estructura de datos se sugiere mantener un arreglo que permita registrar con el indice del vector los numeros que no son primos, a continuacion puedo saber cuales numeros son los primos e informar de manera rapida y eficiente el valor. El gasto de memoria es muy grande, pero es preferible esto a aumentar la complejidad algoritmica.

4. Conclusiones

La teoria matematica, acompaada de la programacion, es una herramienta poderosa si se sabe usar de manera adecuada, es decir, pareciese que es mas facil hallar los numeros primos, pero computacionalmente ocurre todo lo contrario, es mas sencillo hallar los numeros que no lo son, de esta manera encotramos un algoritmo que soluciona el problema por un camino indirecto pero eficiente y correcto.

Por otro lado, vemos que en algunas ocasiones es preferible hacer gastos en memoria que en tiempo, sin embargo no se debe abusar de esta practica.