UNIVERSIDAD DE SANTIAGO DE CHILE
FACULTAD DE CIENCIA
DEPARTAMENTO DE MATEMÁTICA Y CIENCIA DE LA COMPUTACIÓN
LICENCIATURA EN CIENCIA DE LA COMPUTACIÓN

Programación Avanzada
Laboratorio N° 1

Integrantes:

Fernando García
Felipe Parra

Profesor:

Nicolas Theriault

Fecha de entrega:

23 de noviembre de 2018

Índice

1. Introducción

El objetivo principal de este trabajo es implementar algoritmos basicos de aritmetica de polinomios: sumas, restas y multiplicación de dos polinomios.

Los objetivos secundarios son: Implementar algoritmos clasicos de sumas, restas y multiplicacion de polinomios de forma estable.

Analizar y aplicar un algoritmo del tipo dividir y conquistar, especificamente el algoritmo de Karatsuba para la multiplicación de polinomios.
Obtener una implementación eficaz de la multiplicación de polinomios aplicable para polinomios de cualquier grados, optimizada para la multiplicación de polinomios del mismo grado

2. Estructura de Datos Utilizadas

Para los polinomios se creó un struct con nombre polinomio, el cual contiene las variables:

- long* p;
- long g;

long * p: es un arreglo el cual almacenará las constantes. Se eligió que fuera un arreglo ya que hace más sencillo poder saber qué grado acompaña a cada constante.

Long g: es el grado que tendrá el polinomio. Es una buena opción ya que puede tener valores más grandes que un entero normal, lo mejor hubiera sido agregarle const ya que los grados nunca serían negativos y permitiría aún más espacio

3. Formato Utilizado

los polinomios se iniciarán bajo la estructura

$p(x) = a * x^i + b * x^{i+1} + c * x^{i+2} + ...z * x^n$

las constantes son las que serán guardadas en cada espacio de la variable long *p. Esta estructura brinda que de manera automática, al buscar el elemento i del polinomio, el grado de la x que acompañará a esta constante será el valor de i. todos tendrán constantes y aquellos polinomios que no contengan ciertos grados de su x, serán representados como 0 T $x^{(gradoFaltante)}$.

4. Estrategias Utilizadas

Para la multiplicaciónn de polinomios con Reducir y Conquistar, se soluciona el problema rebajando en un grado los polinomios involucrados.
Gracias a la propiedad distributiva de la multiplicación, se obtienen dos nuevos polinomios, cada uno con un grado menor a los originales, y se realiza la multiplicación clasica entre estos.
Una vez realizada la suma y obtenido el nuevo polinomio, este se multiplica por la x que se distribuyó con anterioridad para conseguir el resultado.

En la multiplicacion inductiva con la estrategia de dividir y conquistar se dividio el primer polinomio de grado n en dos polinomios: uno de grado n=2 o menor y otro que contenga los terminos d es posible de implementar con 4 multiplicaciones de polinomios de tama no n=2 mas kn sumas (para algun k).

Para el metodo Karatsuba, se utiliza la misma estrategia utilizada en Dividir y Conquistar, salvo que esta vez se crea una serie de polinomios auxiliares y se aborda la multiplicacion como una suma de factores. El hecho de apoyarse en las sumas para realizar la multiplicacion permite reducir el numero de multiplicaciones (propiamente tales) a realizar.

5. Complejidad

La complejidad de la suma es $O(m+n)$, con m y n grados de los respectivos polinomios, por lo que la complejidad en términos de n es $O(n)$

La complejidad de la resta es $O(m+n)$ con m y n grados de los respectivos polinomios, por lo que la complejidad en términos de n es O(n).

La complejidad de la multiplicación clasica es de la forma O(mn). Si los polinomios son del mismo grado, entonces es O(n^2). Dado que tenemos polinomios con n términos (de grado n -1), se obtendran n^2 productos de coeficientes y $n^2 - (2n-1) = (n-1)^2$ sumas de coeficientes. Con esto podemos concluir que el costo de memoria es de: $n^2 + (2n-1)$ coeficientes. Para reducir la utilización de memoria se ocupa la estrategia reducir y conquistar en una de las entradas. De esta manera, se asociará el producto de polinomios de grado n-1 al producto de polinomios de grado n-2. La recurrencia estará dado por la siguiente fórmula:

$C_n = C_{n-1} + ajustes$

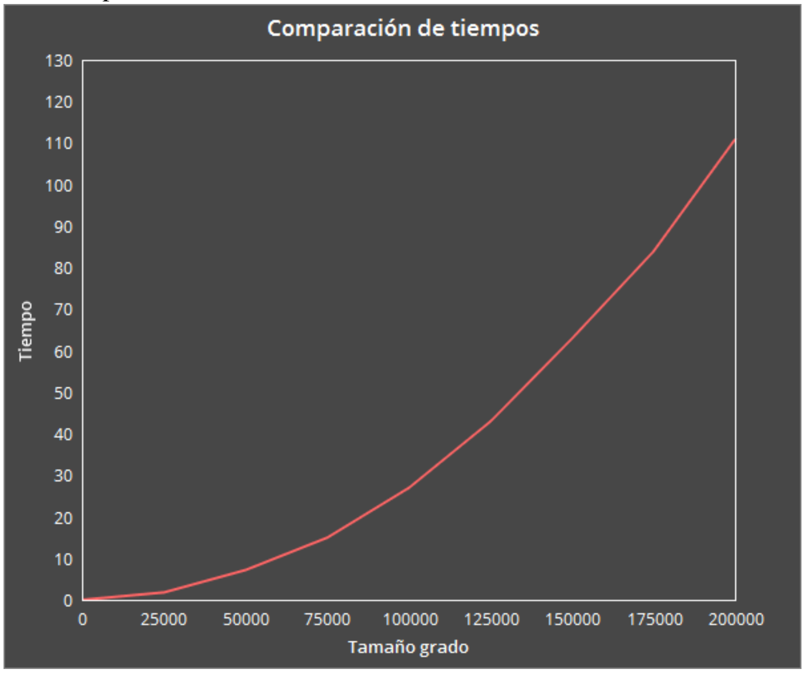
Donde C_n, será el costo de multiplicar dos polinomios de grado n-1. Como resultado se obtendran n^2 multiplicaciones y $(n-1)^2$ sumas de coeficientes.

6. Plataforma Computacional

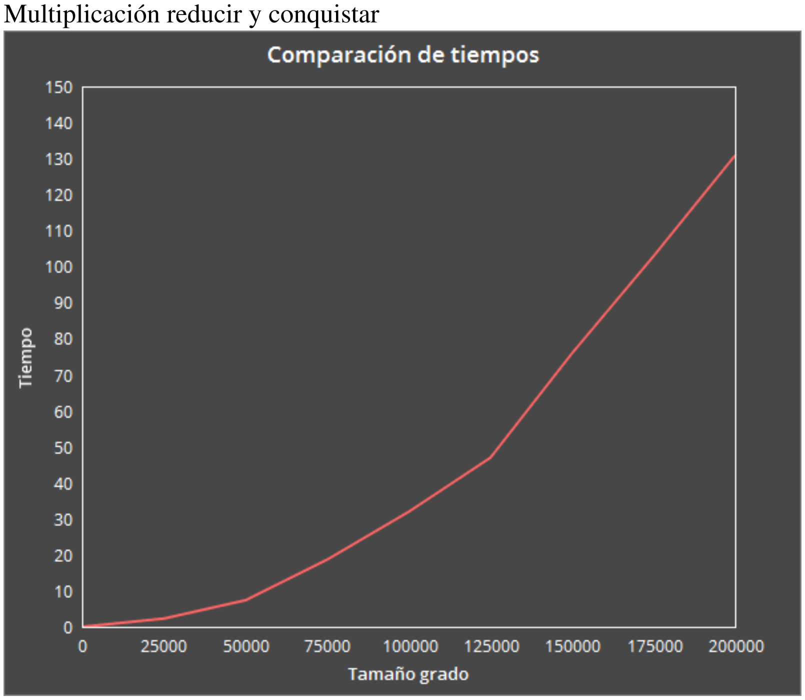
Procesador: Intel i5 7300HQ
Memoria RAM: 8 GB
Sistema Operativo: Ubuntu 18.04
Compilador: GCC 7.2.1
Lenguaje: C11

7. Resultados Experimentales

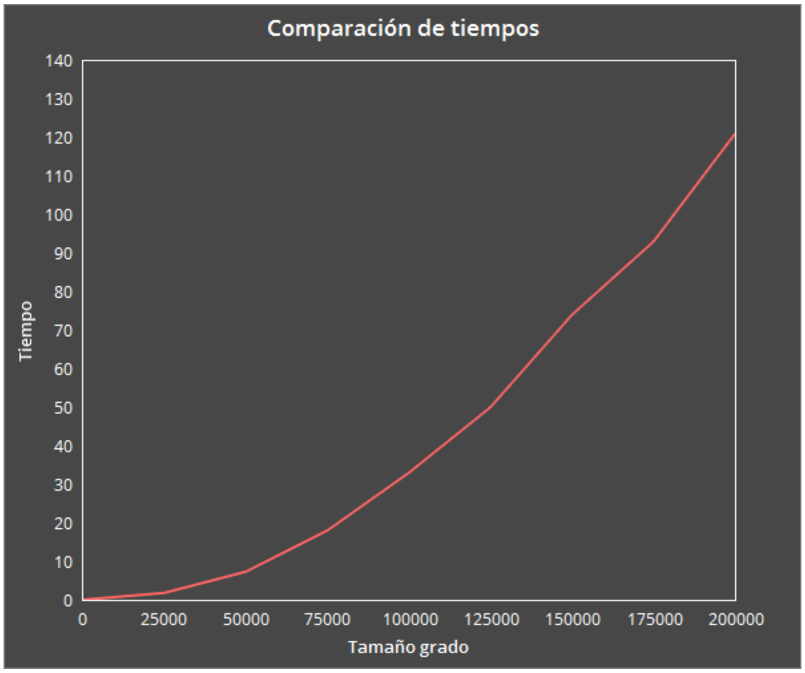
Multiplicación basica



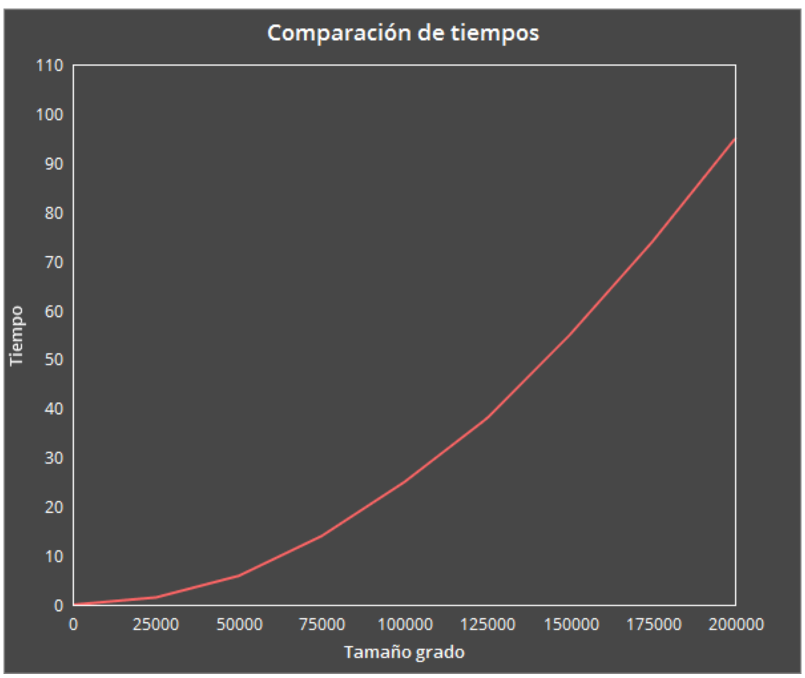
Multiplicación reducir y conquistar



Multiplicación dividir y conquistar



Multiplicacion Karatsuba



Con esto podemos comprobar que la multiplicación de Karatsuba es el método mas eficiente para resolver problemas de multiplicación de polinomios de grados más grandes.