**Método de la ingeniería**

***Fase 1: Identificación del problema***

La empresa especializada en desarrollo de soluciones de nube y locales Oracle desea lanzar una nueva funcionalidad dentro de Java, la cual es *encontrar las raíces de un polinomio*, implementando al menos 2 algoritmos que solucionen esa tarea, debido a que los estudiantes de ingeniería y desarrolladores necesitan de estas funciones básicas para sus programas.

* **Problema:**
  + Dado un polinomio de grado ***n***
  + Encontrar todos los valores de ***x***que hagan a ***P(x) = 0***
* **Requerimientos Funcionales:**

|  |  |
| --- | --- |
| **Nombre** | **R1** |
| **Resumen** | Encontrar las raíces reales de un polinomio de grado n |
| **Entrada** | Polinomio |
| **Salida** | Raíces reales |

|  |  |
| --- | --- |
| **Nombre** | **R2** |
| **Resumen** | Encontrar las raíces imaginarias de un polinomio de grado n |
| **Entrada** | Polinomio |
| **Salida** | Raíces imaginarias |

|  |  |
| --- | --- |
| **Nombre** | **R3** |
| **Resumen** | General polinomio aleatorio de grado 10 |
| **Entrada** | Ninguna |
| **Salida** | Polinomio de grado 10 |

|  |  |
| --- | --- |
| **Nombre** | **R4** |
| **Resumen** | Generar un polinomio |
| **Entrada** | * n -> Grado máximo del polinomio * An -> Constantes del polinomio |
| **Salida** | Polinomio generado |

* **Requerimiento No Funcionales -> Portilla**

***Fase 2: Recopilación de información***

***Definiciones:***

*Polinomio:*

*Un polinomio es una expresión algebraica constituida por la suma o la resta de una serie de términos constantes o variables, los cuales deben ser finitos y pueden tomar exponentes de valores definidos, tomando su exponente se considera el grado del polinomio.*

*Raíces de un polinomio:*

*Se dice que un valor x = a es raíz de un polinomio P(x), cuando al sustituir dicho valor en el polinomio, el resultado es 0; es decir, cuando P(a) = 0*

*Función polinómica:*

*Las funciones polinómicas son, como su nombre lo dice, funciones que constan de un polinomio*.

http://bibliotk.gdl.up.mx/calculo/p10005.jpg

*en donde n es un entero positivo, llamado, grado del polinomio. Resulta evidente, que el coeficiente del grado mayor, no puede ser cero, o sea, a tiene que ser diferente de cero, para que el grado del polinomio se n. Cualquiera de los otros coeficientes puede ser cero.*

***Fase 3: Búsqueda de soluciones creativas***

Para la solución de este problema necesitamos enfocarnos en los algoritmos necesario para hallar las raíces de un polinomio.

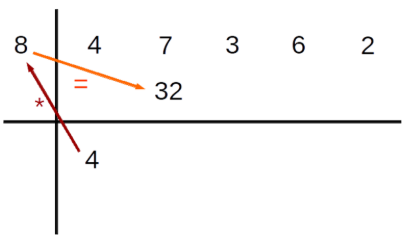
***Alternativa 1:*** Fuerza bruta con el método de Horner.

Esta solución se basa en usar el método de Horner para probar valores particulares de x hasta que P(x) = 0. Aunque el calcular la solución de un polinomio para un determinado valor de x es tarea sencilla, el método de Horner reduce drásticamente la cantidad de operaciones para llegar a ese resultado, esto lo hace un excelente método para buscar las raíces del polinomio probando distintos valores de x.

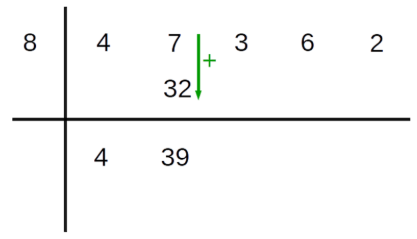
Para explicar cómo funciona el método de Horner vamos a tomar el siguiente Polinomio:

Y lo vamos a evaluar en x = 8, esto debería dar 20210

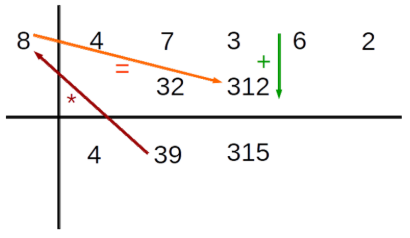
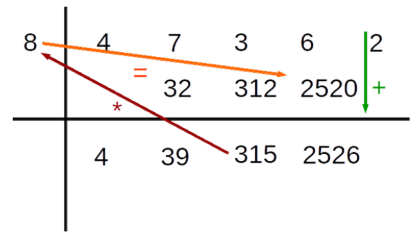
Colocamos los coeficientes del polinomio en una tabla junto con el valor de x que quiere evaluarse. Bajamos el primer coeficiente y lo multiplicamos por el valor de x colocando el resultado debajo del siguiente coeficiente en la tabla



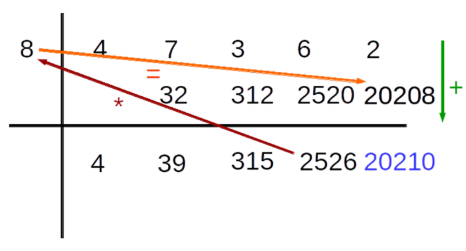
Sumamos los dos valores obteniendo un nuevo resultado parcial



Repetimos la operación para cada coeficiente



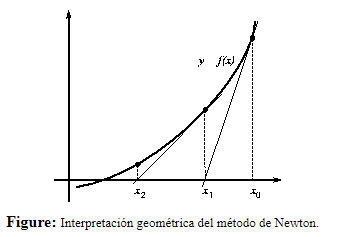
Al llegar al último coeficiente obtenemos el resultado final



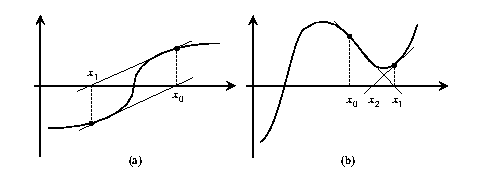
***Alternativa 2:*** Método de Newton para polinomios.

Este método es un algoritmo se usa para encontrar las raíces de un polinomio de cualquiera grado. Este método parte de una aproximación inicial para para la raíz y obtiene una aproximación mejor , la cual se obtiene a partir de la fórmula:

El método de Newton consiste en una lineación del polinomio, es decir, *f* se reemplaza por una recta tal que contiene al punto (*x*0,*f*(*x*0)) y cuya pendiente coincide con la derivada de la función en el punto, *f*'(*x*0). La nueva aproximación a la raíz, *x*1, se obtiene de la intersección de la función linear con el eje *X* de ordenadas.



El método de Newton es muy rápido y eficiente ya que la convergencia es de tipo cuadrático (el número de cifras significativas se duplica en cada iteración). Sin embargo, la convergencia depende en gran medida de la forma que adopta la función en las proximidades del punto de iteración. En la siguiente figura se muestran 2 situaciones en las que el método no es capaz de alcanzar la convergencia o converge a un punto, el cual no es una raíz.



***Alternativa 3:*** Fórmula cuadrática.

La fórmula cuadrática es una manera segura de resolver polinomios de la forma

Colocando los valores a, b y c en la siguiente formula, podremos obtener los valores de para los cuales .

***Alternativa 4:*** Método de Bairstow.

El método de LinBairstow es un método iterativo, basado en el método de Müller y de Newton Raphson. Dado un polinomio  se encuentran dos factores, un polinomio cuadrático . El procedimiento general para el método de Lin Bairstow es:

1. Dado *fn(x)* y r0 y s0
2. Utilizando el método de NR calculamos *f2(x) = x2 – r0x – s0* y *fn-2(x)*, tal que, el residuo de *fn(x)/ f2(x)*sea igual a cero.
3. Se determinan las raíces *f2(x)*, utilizando la formula general.
4. Se calcula *fn-2(x)= fn(x)/ f2(x)*.
5. Hacemos *fn(x)= fn-2(x)*
6. Si el grado del polinomio es mayor que tres regresamos al paso 2
7. Si no terminamos

La principal diferencia de este método, respecto a otros, es que permite calcular todas las raíces de un polinomio (reales e imaginarias).

***Lo demás pronto…***