



Ejemplo Reporte Práctica

POLINOMIO CON COEFICIENTES RACIONALES

Juan Jesús Gutiérrez García | Algoritmos y Estructura de Datos | Sep-2020

Descripción del problema

El objetivo de la práctica es construir un tipo de dato que represente a los polinomios de segundo grado con coeficientes racionales. Un ejemplo de estos puede ser:

$$\frac{3}{4}x^2 + \frac{3}{22}x + \frac{1}{3} = 0$$

En este caso se trata de un polinomio de segundo grado con coeficientes racionales, el objetivo es obtener las raíces del polinomio que esté expresada como una fracción.

Para hacerlo se crearon los siguientes tipos de datos

- Racional, que representa al numerador, denominador y signo de un número racional. Este tipo está definido en racional.h e implementado en racional.c
- Ecseg, es el tipo que representa una ecuación de segundo grado con coeficientes racionales. Además mantiene dos campos o atributos racionales x1 y x2 que almacenarán la solución de la ecuación. Y un entero que es el estado en que se encuentra el número racional:

IMPLEMENTACIÓN DEL RACIONAL

Para implementar el número racional se parte de la siguiente estructura que está definida en el archivo racional.h

```
typedef struct{
    int num; /*numerador, siempre positivo*/
    int den; /*denominador, siempre positivo */
    int sig; /* 0 negativo 1 positivo*/
} racional
```

Tanto el denominador como el numerador siempre serán números positivos, habría que evaluar declararlos como unsigned int. Así el signo del número completo estará en el entero sig.

La función que inicializa el número racional recibirá parámetros enteros con signo y los convertirá a positivos para almacenarlos en la estructura y calculará el valor de sig. Como la fracción siempre se almacena reducida, es decir el $\frac{4}{2}$ se guarda como $\frac{1}{2}$, es necesario calcular el máximo común divisor. Hay dos funciones que se usan para ello uno que verifica los parámetros y la otra que hace el proceso recursivo del cálculo.

Las funciones implementadas son:

- inic_rac, que le da valores iniciales a numerador y denominador, esta función verifica que el denominador no sea cero.
- tostr_rac, convierte un número racional en una representación en cadena para poder ser mostrada con la función printf.
- copia_rac, realiza la copia de un racional en otro.
- comp_rac, compara dos racionales, si el primer parámetro igual con el segundo parámetro.

- e) dist_rac, dice si dos racionales son diferentes
- f) mayq_rac, dice si el primer parámetro es mayor que el segundo
- g) inv_rac, invierte el número racional.
- h) raiz_rac, calcula la raíz cuadrada de un número racional
- i) suma_rac, calcula la suma entre dos racionales.
- j) rest_rac, calcula la resta entre dos racionales.

En el caso de representar el polinomio de segundo grado con coeficientes racionales utilice la siguiente estructura:

```
typedef struct{
    /*Coeficientes*/
    racional a;      /*Variable al cuadrado*/
    racional b;      /*Variable lineal */
    racional c;      /*Constante*/
    /*Raíces de la ecuación*/
    racional x1;
    racional x2;
    /*Estado*/
    int estado; /*Entero que define el estado de la ecuación*/
} ecseg;
```

Para esta estructura cada vez que se modifican los coeficientes se recalculan las raices, y cuando las raices son solicitadas sólo se regresa el valor de x1 o x2. Además la variable estado está planeada para tener los siguientes valores

- 1) #define RAIZ_UNICA, en caso de que se tenga una raiz pero sea igual para x1 y x2
- 2) #define RAIZ_IMG, en caso de no tener raices reales
- 3) #define RAIZ_DOBLE si x1 y x2 tienen racionales diferentes que sean raices del polinomio.
- 4) #define RAIZ_ERR, en caso de que los coeficientes no cumplan con ser un polinomio de segundo grado, esto es que el coeficiente a sea cero.

El proyecto no está terminado porque no me dio tiempo, así que le envío este avance esperando me de una semana más para terminarlo.

CONCLUSIONES

Ya probé el funcionamiento correcto de los racionales, para ello use la función main que está en el archivo usarac.c y no encontré que tenga errores.

Hice algunas pruebas para el polinomio de racionales, que están en el archivo usarac(1).c.

Una de mis principales dudas es al calcular la raíz de un número racional y no se obtenga como resultado un racional, por ejemplo

$$\sqrt{\frac{1}{2}}$$

En este caso el numerador tiene raíz 1 pero el denominador tiene raíz no exacta... ¿cómo debo procesar eso?

Espero si me pueda dar el tiempo solicitado.... Sólo una semana.

Gracias