

Estructura de datos proyecto(reporte de complejidad)

Rui Yu Lei Wu

Código: 8978227

La implementación elegida para el TAD BigInteger fue de vectores de STL.

Constructores:

En este caso el TAD BigInteger cuenta con tres constructores.

1. Constructor por defecto: Este constructor inicializa un vector vacío denominado vec. La complejidad de esta operación es $O(1)$, debido a que inicializar un vector tiene coste constante.
2. Constructor String: Este constructor inicializa un vector el cual contiene en la posición 0 el número 1 o 0 correspondiente al signo del número, seguido de eso cada posición corresponde a un número “n”, en el cual cada número representa a un número del string. En este caso el vector alberga el numero dado la vuelta. La complejidad de esta operación es $O(n)$ donde n es el número de elementos en el string.
3. Constructor copia: Este constructor copia los elementos de un BigInteger y los alberga en otro nuevo, esto sin modificar el BigInteger original. La complejidad de esta operación es $O(n)$ donde n representa el número de elementos del BigInteger original, esto debido a que lo único que hace este constructor es copiar los elementos de un vector y copiarlos a otro.

Operaciones:

El TAD BigInteger cuenta con seis operaciones principales.

1. Operación add: Esta operación es la encargada de sumar dos números de n dígitos, esta operación contiene un ciclo for que va desde i hasta el final del BigInteger más pequeño, seguido de esto si el tamaño de ambos BigIntegers es diferente entra en otro ciclo que itera hasta que el residuo sea igual a cero y i igual al tamaño del BigInteger más grande. En este caso la complejidad de esta implementación es $O(n)$ donde n es el tamaño del BigInteger más grande.
2. Operación subtract: Esta operación es la encargada de restar dos números de n dígitos, esta operación tiene un concepto muy parecido a la operación add, un ciclo que va desde i hasta el tamaño del BigInteger con menos dígitos, y si el tamaño de ambos BigInteger es diferente entra en otro ciclo el cual termina cuando i es igual al tamaño del BigInteger más grande. La complejidad de esta implementación al igual que la operación add es $O(n)$ donde n es el tamaño del BigInteger más grande.
3. Operación Product: