Ferdiansyah Irawan

2111083008

TRPL 3A

Program StaticHuffmanCompression

Merupakan implementasi kompresi dan dekompresi data menggunakan Java dengan menggunakan GZIP (Gnu Zipped File) sebagai format kompresi.

Metode compressData:

- Metode ini menerima data dalam bentuk array byte (byte[] data).

- Sehingga membuat ByteArrayOutputStream yang akan digunakan untuk menampung data yang sudah dikompresi.

- Membungkus ByteArrayOutputStream dalam GZIPOutputStream.

- Menulis data ke GZIPOutputStream, yang akan mengkompresi data.

- Mengembalikan data yang sudah dikompresi sebagai array byte.

Metode decompressData:

- Metode ini menerima data yang sudah dikompresi dalam bentuk array byte (byte[] compressedData).

- Ini membuat ByteArrayInputStream dari data yang sudah dikompresi.

- Membuat ByteArrayOutputStream untuk menampung data yang sudah didekompresi.

- Membungkus ByteArrayInputStream dalam GZIPInputStream untuk dekompresi.

- Membaca data dari GZIPInputStream, men-dekompresinya, dan menulisnya ke ByteArrayOutputStream.

- Mengembalikan data yang sudah didekompresi sebagai array byte.

Metode main:

- Berisikan data teks yang ingin dikompresi didefinisikan sebagai string (originalData).

- Data string ini dikonversi menjadi array byte (dataBytes) menggunakan getBytes() method.

- Kemudian, data tersebut dikompresi dengan memanggil compressData dan hasilnya dicetak ke layar sebagai string.

- Data yang sudah dikompresi kemudian didekompresi dengan memanggil decompressData, dan hasil dekompresi dicetak ke layar.

Try-Catch:

Program ini menggunakan blok try-catch untuk menangani pengecualian yang mungkin muncul saat mengompresi atau mendekompresi data. Jika terjadi kesalahan, pengecualian akan ditangkap dan pesan kesalahan akan dicetak ke layar.

Program ini mengilustrasikan cara menggunakan kompresi GZIP dalam Java untuk mengurangi ukuran data yang akan disimpan atau dikirim melalui jaringan. Ini berguna terutama saat Anda perlu mengompresi data sebelum menyimpannya di disk atau mengirimnya melalui jaringan untuk menghemat bandwidth.

Penggunaan kelas-kelas GZIPOutputStream dan GZIPInputStream yang disediakan oleh Java untuk melakukan kompresi dan dekompresi data dalam format GZIP. Data asli dibaca, dikompresi, dan kemudian dikirim ke aliran keluaran. Ketika data perlu didekompresi, data yang sudah dikompresi dibaca, didekompresi, dan hasil dekompresi ditulis ke aliran keluaran. Selanjutnya, program ini menggunakan array byte untuk menyimpan dan memanipulasi data dalam bentuk biner.