

Nama : Azhar Adi Dirgantara Setiawan

Npm : 06.2020.1.07277

UAS JARINGAN MULTIMEDIA

Sourcecode

```
#include <iostream>

#include <string>

#include <sstream>

using namespace std;

// Kompresi menggunakan metode Run-Length

string compress(const string& str) {

    string compressed;

    int count = 1;

    for (int i = 1; i < str.length(); i++) {

        if (str[i] == str[i - 1]) {

            count++;

        } else {

            stringstream ss;

            ss << count;

            compressed += str[i - 1] + ss.str();

            count = 1;

        }

    }

    // Menambahkan karakter terakhir dan jumlahnya

    stringstream ss;

    ss << count;

    compressed += str[str.length() - 1] + ss.str();

    return compressed;
```

```
}
```

```
// Dekompresi menggunakan metode Run-Length
```

```
string decompress(const string& str) {
```

```
    string decompressed;
```

```
    int i = 0;
```

```
    while (i < str.length()) {
```

```
        char character = str[i];
```

```
        i++;
```

```
        int count = 0;
```

```
        while (i < str.length() && isdigit(str[i])) {
```

```
            count = count * 10 + (str[i] - '0');
```

```
            i++;
```

```
        }
```

```
        decompressed.append(count, character);
```

```
    }
```

```
    return decompressed;
```

```
}
```

```
int main() {
```

```
    string inputString;
```

```
    cout << "Enter a string: ";
```

```
    getline(cin, inputString);
```

```
    string compressedString = compress(inputString);
```

```
    cout << "Compressed string: " << compressedString << endl;
```

```
    string decompressedString = decompress(compressedString);
```

```

cout << "Decompressed string: " << decompressedString << endl;

return 0;
}

```

Hasil Run

```

Enter a string: aakkku aanakk baiiiiikk
Compressed string: a2k3u1 1a2n1a1k2 1b1a1i5k2
Decompressed string: aakkku aanakk baiiiiikk

-----
Process exited after 14.08 seconds with return value 0
Press any key to continue . . .

```

Hasil kompresi string yang Anda berikan tampaknya didasarkan pada konsep pengkodean run-length encoding. Metode ini mengompresi string dengan cara mencatat berapa kali setiap karakter berulang secara berurutan.

Dalam string "aakkku aanakk baiiiiikk", berikut adalah langkah-langkahnya:

1. Perhatikan karakter pertama, 'a'. Kemudian kita hitung berapa kali 'a' berulang secara berurutan. Dalam kasus ini, 'a' berulang dua kali, jadi kita menuliskannya sebagai "a2".
2. Karakter berikutnya adalah 'k'. 'k' hanya muncul satu kali, jadi kita menuliskannya sebagai "k1".
3. Karakter selanjutnya adalah 'k' lagi. Karena 'k' berulang dua kali, kita menuliskannya sebagai "k2".
4. Karakter berikutnya adalah 'u'. 'u' hanya muncul satu kali, jadi kita menuliskannya sebagai "u1".
5. Karakter selanjutnya adalah ' '. Karena spasi tidak berulang, kita tidak perlu mencatatnya.
6. Karakter berikutnya adalah 'a'. 'a' muncul satu kali, jadi kita menuliskannya sebagai "a1".
7. Karakter selanjutnya adalah 'a' lagi. Karena 'a' berulang dua kali, kita menuliskannya sebagai "a2".
8. Karakter berikutnya adalah 'n'. 'n' muncul satu kali, jadi kita menuliskannya sebagai "n1".
9. Karakter selanjutnya adalah 'a'. 'a' muncul satu kali, jadi kita menuliskannya sebagai "a1".
10. Karakter berikutnya adalah 'k'. 'k' muncul dua kali, jadi kita menuliskannya sebagai "k2".
11. Karakter selanjutnya adalah spasi lagi. Kita tidak perlu mencatatnya.
12. Karakter berikutnya adalah 'b'. 'b' muncul satu kali, jadi kita menuliskannya sebagai "b1".
13. Karakter selanjutnya adalah 'a' lagi. 'a' muncul satu kali, jadi kita menuliskannya sebagai "a1".
14. Karakter berikutnya adalah 'i'. 'i' muncul lima kali, jadi kita menuliskannya sebagai "i5".
15. Karakter selanjutnya adalah 'k' lagi. 'k' muncul dua kali, jadi kita menuliskannya sebagai "k2". Jadi, hasil kompresi string "aakkku aanakk baiiiiikk" menjadi "a2k3u1 1a2n1a1k2 1b1a1i5k2".