

Codice C++: Conversione di un messaggio in binario

Gruppo 5

February 5, 2025

1 Introduzione

Il seguente codice in C++ implementa un sistema per convertire un messaggio di testo in binario e viceversa, scrivendolo poi su file.

2 Codice sorgente

```
#include <iostream>
#include <string>
#include <fstream>
#include <bitset>
#include <sstream>

#include "funzioni.h"

using namespace std;

IpHeader datagram() {
    IpHeader header;
    header.Version = "0100"; // IPv4 (4)
    header.HLen = "0101"; // 5 words x 4 = 20 bytes (5)
    header.TOS = "00000000"; // Tipo di servizio di default (0)
    header.TotalLength = "00000000_00000000"; // Lunghezza totale
        del pacchetto (da calcolare dinamicamente)
    header.Identification = "00000000_00000000"; // Dinamico in
        base al pacchetto
    header.Flags = "010"; // DF (Don't Fragment) settato (2)
    header.FragmentOffset = "000000000000"; // Nessuna
        frammentazione di default (0)
    header.TTL = "01000000"; // Predefinito per Linux (su Windows
        128) (64)
    header.Protocol = "00000110"; // Predefinito in TCP (pu
        essere cambiato) (6)
    header.HeaderChecksum = "00000000_00000000"; // Calcolato
        dinamicamente
    header.SourceIpAddress = "00000000_00000000_00000000_00000000";
        // Indirizzo di provenienza (da specificare)
```

```

        header.DestinationIpAddress = "00000000_00000000_00000000_
            00000000"; // Indirizzo di destinazione (da specificare)
        header.Options = "00000000"; // Nessuna opzione di default (0)
        header.Padding = "00000000"; // Padding di default (0)

        return header;
    }

    string leggiFile(){
        ifstream fileIn("messaggio.txt"); // Apro file 'messaggio.txt'
            in lettura
        string riga;
        getline(fileIn, riga); // Copio prima riga del file in 'riga'
        fileIn.close();
        return riga;
    }

    void scriviFile(string testo){
        ofstream fileOut("frame.txt"); // Apro il file 'frame.txt' in
            scrittura
        if(!fileOut) {
            cout << "Errore_nell'apertura_del_file." << endl;
        }
        fileOut << testo; // Scrivo il contenuto della variabile 'testo'
            nel file
        fileOut.close(); // Chiudo il file
    }

    string chatToBin(string carattere){
        string binario = "";
        binario += bitset<8>(carattere).to_string() + "_";
        return binario;
    }

    string stringaABinario(string testo){
        string binario = "";
        for(char carattere : testo) {
            binario += bitset<8>(carattere).to_string() + "_";
        }
        return binario;
    }

    string binarioAStringa(string input) {
        stringstream lettoreBinario(input); // Crea uno stream per
            elaborare l'input
        string carattereBinario;
        string testo;

        // Legge ogni gruppo di 8 bit (separati da spazi) e li converte
            in caratteri
        while (lettoreBinario >> carattereBinario) {
            char carattere = bitset<8>(carattereBinario).to_ulong(); //
                Converte il binario in un carattere
            testo += carattere; // Aggiunge il carattere alla stringa
                risultante
        }
        return testo;
    }

```

```

}

int main(){
    string testo = leggiFile(); // Creo 'testo' e ci metto il
        contenuto di 'messaggio.txt'
    string binario = stringaABinario(testo); // Converto il
        contenuto di 'testo' in binario e lo salvo in 'binario'
    cout << "Messaggio_in_binario:" << binario << endl; // Stampo
        binario a schermo
    scriviFile(binario); // Scrivo 'binario' in 'frame.txt'
}

```