



```
#include <iostream>
#include <ostream>

using namespace std;

struct Data { int dia, mes, ano; };

ostream& operator<< (ostream &saida, Data d)
{
    // ( ( ( saida << d.dia) << '/' ) << d.mes ) << '/' ) << d.ano;
    //
    // ( ( (1 + 2) + 3 ) + 4 ) + ...

    saida << d.dia << '/' << d.mes << '/' << d.ano;

    return saida;

    // Ou: return saida << d.dia << '/' << d.mes << '/' << d.ano;
}

int main ()
{
    Data hoje {10, 5, 2023}; cout << "Hoje: " << hoje << '\n';
}
```

```
#include <fstream>
#include <iostream>

using namespace std;

int main ()
{
    cout << "Nome do arquivo: "; string nome; cin >> nome;

    ifstream arq(nome.data()); // Modo texto (padrão).

    if (not arq.is_open()) { cerr << "Erro!\n"; return 1; }

    char c = arq.get();

    if (arq.eof()) { cout << "Arquivo vazio!\n"; return 0; }
    else cout << "Primeiro caractere: [" << c << "]\n";

    c = arq.get();

    if (arq.eof()) cout << "Arquivo com um caractere apenas\n";
    else cout << "Segundo caractere: [" << c << "]\n";
}
```

```
#include <fstream>
#include <iostream>
#include <limits>

using namespace std;

int main ()
{
    cout << "Nome de um arquivo qualquer: "; string nome; cin >> nome;

    ifstream arq (nome.data(), std::ios_base::in | std::ios_base::binary);

    cout << "char tem sinal? " << numeric_limits<char>::is_signed << '\n';

    unsigned char byte = arq.get(); // Se usar char, pode vir negativo...
    // ou seja, será ruim como índice.

    if (arq.eof()) { cout << "Vazio!\n"; return 0; }

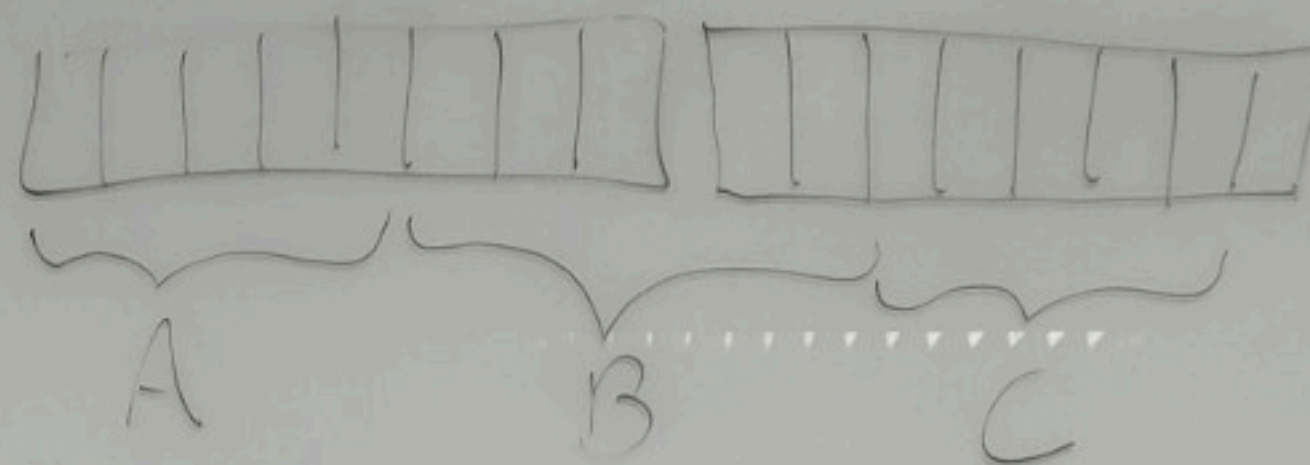
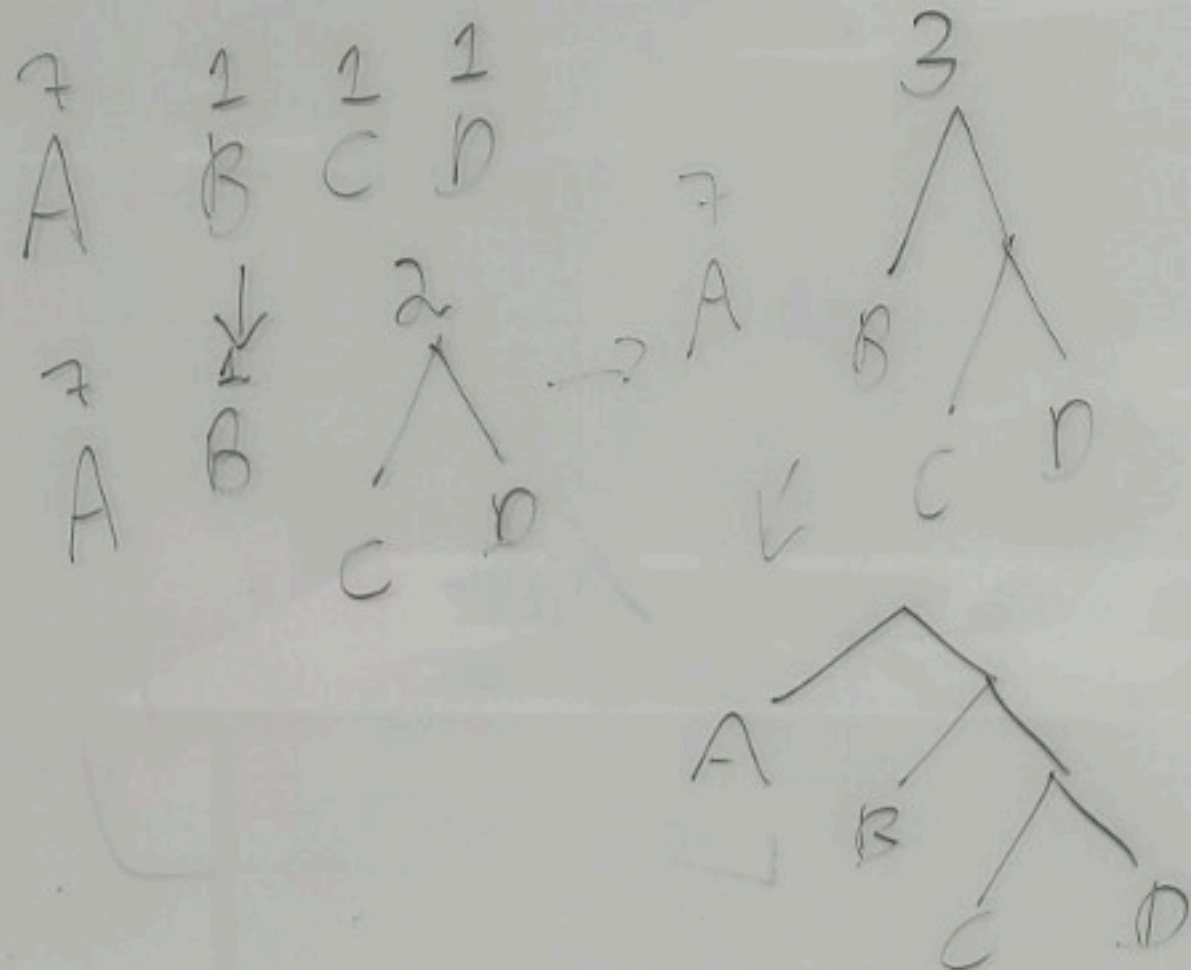
    cout << "Lido o byte " << (int) byte << '\n';
}
```


ASCII

a ~> 1+

! ~>

F ~> 65+5



"ABCDAAAAAA"

A's: 7 bits
B: 2
C: 3
D: 3
15 bits

A -> 00

B -> 01

C -> 10

D -> 11

