

Exemplo 1 - Lista linear encadeada.cpp

```
#include <iostream>
#include <string.h>
#include <locale.h>
using namespace std;

struct DADOS_ALUNO{
    int CodAluno;
    char Nome[100];
    int Turma;

    struct DADOS_ALUNO *ptrLink;
};

int main(){
    setlocale(LC_ALL, "");
    int Ind;
    struct DADOS_ALUNO *ptrCabeca;
    struct DADOS_ALUNO *ptrPrimeiraNo, *ptrSegundoNo;
    struct DADOS_ALUNO *ptrAux;
    ptrCabeca = new DADOS_ALUNO;
    ptrCabeca->ptrLink = NULL;

    // PRIMEIRO NÓ: Cria um novo nó
    ptrPrimeiraNo = new DADOS_ALUNO;
    ptrPrimeiraNo->CodAluno = 10;
    strcpy(ptrPrimeiraNo->Nome, "José");
    ptrPrimeiraNo->Turma = 250;
    ptrPrimeiraNo->ptrLink = NULL;
    // Liga o primeiro nó ao nó cabeça
    ptrCabeca->ptrLink = ptrPrimeiraNo;

    // SEGUNDO NÓ: Cria um novo nó
    ptrSegundoNo = new DADOS_ALUNO;
    ptrSegundoNo->CodAluno = 20;
    strcpy(ptrSegundoNo->Nome, "Maria");
    ptrSegundoNo->Turma = 250;
    ptrSegundoNo->ptrLink = NULL;

    // Liga o segundo nó ao primeiro nó
    ptrPrimeiraNo->ptrLink = ptrSegundoNo;
    ptrAux = ptrCabeca->ptrLink;
    Ind = 1;

    while(ptrAux != NULL){
        cout << "Nó: " << Ind << endl;
        cout << "Código do Aluno: " << ptrAux->CodAluno << endl;
        cout << "Nome: " << ptrAux->Nome << endl;
```

```

        Exemplo 1 - Lista linear encadeada.cpp
        cout << "Turma: " << ptrAux->Turma << endl;
        cout << endl << endl;
        Ind++;
        ptrAux = ptrAux->ptrLink;
    }
    // Libera o espaço de memória
    delete ptrCabeca;
    delete ptrPrimeiraNo;
    delete ptrSegundoNo;
}

```