	<b>ROTEIRO DE PRÁTICA EM LABORATÓRIO</b> ESTRUTURA DE DADOS – 2019.1 PROF. ANDERSON LIMA	
<b>ATIVIDADE</b>	LISTA SEQUENCIAL ESTÁTICA	<b>DATA</b> 12/03/2019

## OBJETIVO

Implementar as principais operações complementares de uma lista sequencial estática.

## PREPARAÇÃO

- I. Abrir o **CodeBlocks**.
- II. Abrir o projeto criado na aula anterior (**projeto-lista**).

## ROTEIRO

1. Conferir se o arquivo de header possui a declaração das operações usadas no exercício anterior. (*ListaSequencial.h*)
2. Abrir o arquivo de implementação (*ListaSequencial.c*)
3. Conferir se o arquivo possui a implementação das operações básicas implementadas na aula anterior.
4. Alterar o arquivo **main.c** para testar a estrutura de dados do tipo lista.
5. Criar no arquivo de *header* e de implementação a função para imprimir o conteúdo da lista.

### **ListaSequencial.h**

```
// Impressao da lista
void imprime_lista(Lista* li);
```

### **ListaSequencial.c**

```
// Impressao da lista
void imprime_lista(Lista* li) {
    printf("\n\n");

    if (li == NULL) {
        printf("Lista nao existe!\n");
        return;
    }

    if (li->qtd == 0) {
        printf("Lista vazia!\n");
        return;
    }
}
```



## ROTEIRO DE PRÁTICA EM LABORATÓRIO

ESTRUTURA DE DADOS – 2019.1

PROF. ANDERSON LIMA

### ATIVIDADE

LISTA SEQUENCIAL ESTÁTICA

### DATA

12/03/2019

```
printf("=====\n");
printf("CONTEUDO DA LISTA SEQUENCIAL ESTATICA \n");
printf("-----\n");
for (int i = 0; i < li->qtd; i++) {
    printf("POS %d ->| %d %s %.1f %.1f %.1f |<- \n",
        i,
        li->dados[i].matricula,
        li->dados[i].nome,
        li->dados[i].n1,
        li->dados[i].n2,
        li->dados[i].n3);
}
printf("-----\n");
printf("Tamanho: %d \n", li->qtd);
printf("-----\n");
}
```

6. Criar no arquivo de testes a função para efetuar a leitura dos dados de um novo aluno.

**main.c**

```
#include (...)

void cadastrar_aluno(struct aluno *novoAluno) {
    printf(" Informe a matricula: ");
    scanf("%d%c", &novoAluno->matricula);


    printf(" Informe o nome: ");
    scanf("%s%c", &novoAluno->nome);

    printf(" Informe a 1a nota: ");
    scanf("%f%c", &novoAluno->n1);

    printf(" Informe a 2a nota: ");
    scanf("%f%c", &novoAluno->n2);

    printf(" Informe a 3a nota: ");
    scanf("%f%c", &novoAluno->n3);
}

int main() {
    (...)
}
```


	<b>ROTEIRO DE PRÁTICA EM LABORATÓRIO</b> ESTRUTURA DE DADOS – 2019.1 PROF. ANDERSON LIMA		
<b>ATIVIDADE</b>	LISTA SEQUENCIAL ESTÁTICA	<b>DATA</b>	12/03/2019

7. Incluir no arquivo de testes (**main.c**) o trecho de código para exibir o menu do sistema ao usuário.

**main.c**

```
int main() {
    (...)
    int opcao = 0;
    struct aluno *novo = NULL;
    do {
        printf("\n\n");
        printf("----- \n");
        printf("1: Inserir aluno \n");
        printf("2: Imprimir lista \n");
        printf("0: Sair \n");
        printf("----- \n");
        printf("Informe a opcao desejada: ");
        scanf("%d", &opcao);
        switch (opcao) {
            case 1:
                novo = (struct aluno*) malloc(sizeof(struct aluno));
                cadastrar_aluno(novo);
                insere_lista_final(listaAluno, *novo);
                break;
            case 2:
                imprime_lista(listaAluno);
                break;
            case 0:
                break;
            default:
                printf("Opcao invalida! %d\n", opcao);
                break;
        }
    } while (opcao != 0);
    return 0;
}
```

8. Implemente a rotina **insere\_lista\_inicio** e crie no menu a opção “3 – Inserir aluno no início” que chame a rotina para ler os dados de um novo aluno e insira o aluno no início da lista.
9. Crie no menu a opção “4 – Remover aluno por matrícula”, que permita ao usuário informar uma matrícula e excluir da lista o aluno com a matrícula informada. Caso não exista um aluno com a matrícula, o sistema deve exibir uma mensagem apropriada.

	<b>ROTEIRO DE PRÁTICA EM LABORATÓRIO</b> ESTRUTURA DE DADOS – 2019.1 PROF. ANDERSON LIMA		
<b>ATIVIDADE</b>	LISTA SEQUENCIAL ESTÁTICA	<b>DATA</b>	12/03/2019

10. Implemente a rotina **insere\_lista\_ordenada** e altere a opção “1 – Inserir aluno” para chamar essa rotina para inclusão de novos alunos na lista.
11. Implemente a rotina **busca\_lista\_mat** e crie no menu a opção “5 – Pesquisar aluno por matrícula”, que permita ao usuário informar uma matrícula e exibir os dados do aluno com a matrícula informada. Caso não exista um aluno com a matrícula, o sistema deve exibir uma mensagem apropriada.