



1º trabalho prático

Objetivo

Consolidar os conceitos de listas dinamicamente encadeadas, por meio da aplicação dessa estrutura e suas variações na resolução de um problema.

Instruções

1. O trabalho pode ser desenvolvido individualmente ou em equipe de, **no máximo, duas** pessoas.
2. Data de entrega: **05/12/19**.
3. O trabalho deve ser desenvolvido na linguagem C.
 - Deve ser implementado o conceito de TAD, portanto, devem ser implementados:
 - o arquivo de **declaração** dos subprogramas – *header* (.h)
 - o arquivo com a **definição** dos subprogramas declarados no *header* (.c)
 - e o arquivo contendo o programa principal (.c).
 - Os dois últimos arquivos (.c) devem incluir chamada ao arquivo .h.
4. Os arquivos do trabalho devem ser compactados em um único arquivo que deve ser nomeado de acordo com o nome dos integrantes da equipe seguido de seu R.A.
 - Exemplos:
 - NomeAluno123456.zip
 - FulanodeTal123456_OutroSicrano678910.zip.
5. O trabalho deve ser entregue via Moodle, por meio de um link para *upload* de arquivos disponível até as 23:55 h da data da entrega.
6. É permitida a interação entre as equipes, mas **cópias** de códigos implicará automaticamente na atribuição de **nota zero a todos** os trabalhos envolvidos.
7. A ordem de apresentação dos trabalhos se dará por ordem alfabética de acordo com os nomes dos integrantes das equipes.

Descrição do trabalho

1. Faça um programa que armazene em uma **lista dinâmica simplesmente encadeada** os seguintes dados de aluno: nome, r.a., nota 01, nota 02 e média. O programa deve possuir as seguintes funcionalidades:
 - a) Inserir novo aluno no final da lista: fornecer somente nome, r.a., nota 01 e nota 02;
 - b) Inserir novo aluno no início da lista: fornecer somente nome, r.a., nota 01 e nota 02;
 - c) Calcular a média aritmética para cada um dos alunos, a partir da nota 01 e nota 02.
 - d) Buscar o aluno com a maior média;
 - e) Alterar os dados de um aluno – exceto a média, que deve ser recalculada quando uma das notas for modificada;
 - f) Mostrar todos os alunos cadastrados;
 - g) Mostrar os alunos aprovados, isto é, com média maior ou igual a 6.0;
 - h) Mostrar os alunos reprovados, isto é, com média menor a 6.0.
 - i) Excluir um aluno.



2. Faça um programa que armazene valores inteiros em uma **lista circular duplamente encadeada**, de maneira ordenada. O programa deve possuir as seguintes funcionalidades:
- Inserir ordenado: procurar a posição que o elemento deve ser inserido de modo que todos os elementos da lista permaneçam em ordem crescente.
 - Mostrar a lista circular duplamente encadeada;
 - Excluir um número da lista.

O que será avaliado

- Implementação correta das listas e respectivas operações conforme a descrição deste trabalho.
- Organização do código (uso correto do conceito de TAD, endentação, coesão dos subprogramas).
 - coesão dos subprogramas = o subprograma tem um único propósito; desse modo, fazer com que o programa principal contenha a maior parte das mensagens de interação com o usuário.
- Usabilidade do programa:
 - facilidade de utilização proporcionada pelas telas (menus do console) de interação com o usuário;
 - mensagens de erro para situações como: lista vazia, valores inválidos, etc.