

Tipos Abstratos de Dados (TADs)

1. Você deve alterar o TAD `ListaVetorInteiros` implementado durante a aula (slides sobre TADs disponíveis no Classroom) para que ele suporte duas novas operações relacionadas à remoção de elementos da lista de inteiros: `remover_primeiro` e `remover_ultimo`, e uma terceira operação `inverte` que modifica a ordem dos elementos da lista (ou seja `[1,2,3]` para `[3,2,1]`).
2. Implemente um TAD `Data` com três atributos inteiros: `dia`, `mês` e `ano`. Faça um construtor que inicializa as três variáveis e suponha que os valores passados serão sempre corretos. O TAD deve possuir um método para exibir a data em formato de números separados por barra: `dia/mes/ano` e outro método para exibir a data por extenso (ex: 28 de fevereiro de 2021). Por fim, você deve implementar um método `getDataEmSegundos` que recebe uma data como argumento e retorna sua representação em segundos (em caso de dúvidas, buscar por *unix timestamp* - vale lembrar que a data inicial considerada será 01/01/1970 e que para fins de simplificação não vamos considerar o horário no cálculo, o padrão será sempre 00:00:00).
3. Implemente um TAD `Aluno` que contenha minimamente os seguintes atributos: `nome`, `matrícula`, `notas`, `media_notas`. É importante ressaltar que o atributo `notas` está relacionado à quantidade de matérias cursadas pelo aluno. Ou seja, se o aluno cursa 10 matérias, o programa deve armazenar às 10 notas equivalentes (1 nota para cada matéria). Você deve implementar um construtor para inicialização das variáveis. No caso do atributo `media_notas`, ele deve ser inicializado com 0 (zero) até os respectivos valores de notas sejam lidos. Você deve implementar um método `adicionarNota` responsável pelo preenchimento do vetor de notas do aluno – neste caso você pode utilizar uma variável auxiliar para controle dos índices do vetor de notas. Além disso, você deve implementar um método `calcularMediaAluno`, que recebe como argumento as notas e retorna a média dos valores. Por fim, você deve implementar um método `imprimeInformacoesAluno`, que imprime as informações do aluno.

Considerações!

- Todos os exercícios devem conter `.h`, `.cpp`, e `main.cpp`;
- O seu `main.cpp` deve conter, minimamente, instruções para criação (instanciação de objetos) e chamadas das funções implementadas (TODAS!!!).