

Aula Prática 02 – Alocação Dinâmica e TADs

Data de entrega: 09/09/2018 até 22:00.

Atenção: *O que vale é o horário do RunCodes, e não do seu, ou do meu, relógio.*

Procedimento para a entrega:

1. Para cada questão, implemente sua solução.
 2. Para testar suas soluções, implemente um único método *main()*, que poderá conter, por exemplo, um *Menu* de interações e possibilidades do usuário especificar os dados de entrada.
 3. Especifique o *Makefile* com as instruções necessárias para compilação e execução do seu código, sendo que o *Makefile* deve conter também o redirecionamento da entrada e da saída (e.g., **`./prog.exe < input.txt > output.txt`**.)
 4. Compacte em um único diretório o seu código fonte juntamente com o *Makefile*, o arquivo de entrada e o arquivo de saída usados para testes (e.g., *input.txt*, *output.txt*).
 5. Faça a entrega do arquivo compactado, obrigatoriamente em formato *.zip*, no *RunCodes*, na tarefa correspondente.
- Não utilize caracteres acentuados ou especiais para nomes de pastas, arquivos e na especificação de comentários no código.
 - Implemente em conformidade com boas práticas para reuso e modularização do código.

- Bom trabalho!

Questão 01

Considere um sistema de matrícula que inicialmente deve ser composto pelos seguintes Tipos Abstratos de Dados (TADs): *aluno*, representado por seu nome, CPF, número de matrícula, curso e data de ingresso; *disciplina*, representada por seu nome, código, carga-horária (4 ou 8 horas), curso e sala em que é oferecida; *professor*, representado por seu CPF, nome e departamento onde esta lotado. Especifique e implemente, utilizando conceitos de TADs e alocação dinâmica de memória, uma versão simples desse sistema de matrícula, em que seja possível inserir/remover/atualizar/buscar alunos, disciplinas e professores. O sistema deverá ainda permitir a listagem de todos os professores, os alunos e as disciplinas alocados, exibindo todos os detalhes destes elementos.