

Laboratório 3: Tipo Abstrato de Dados

Objetivo: Revisar conceitos de ponteiros e alocação dinâmica. Capacitar ao aluno trabalhar com a especificação e implementação independente (em arquivos separados) de Tipos Abstratos de Dados (TAD) simples.

Exercícios:

1. Implementar o exercício 7 do exemplo *Manipula_Pares* visto em sala de aula (versão UPM), com manipulação de structs e alocação dinâmica. Sugestão: implemente a versão do exercício 6 primeiro.
2. Implementar o TAD dos números racionais definido na aula teórica, utilizando um projeto com módulos independentes, para o arquivo de especificação da TAD e o arquivo da aplicação do usuário. Para isso, gerar um arquivo racional.c com a implementação das operações do TAD, um arquivo cabeçalho racional.h com a definição da estrutura e das operações do TAD (protótipos) e um arquivo usuario.c, contendo o programa principal que manipulará o TAD.
 - a. Implemente, compile e execute o projeto utilizando o arquivo usuário da forma como está descrito no material da aula teórica: o arquivo usuário.c lê dois números racionais (digitados pelo usuário) e calcula um novo racional definido pela soma dos dois anteriores, apresentando o resultado desse novo racional para o usuário.
 - b. Incremente a TAD Racional, incluindo a operação `prod()`, que realiza o produto de dois números racionais. Teste a nova operação, alterando o arquivo usuário.c.
 - c. Alterar o arquivo usuário.c de forma que sejam utilizadas as operações disponibilizadas no TAD para resolver as expressões abaixo, imprimindo o resultado de cada expressão. Obs: Alocar e ler os 8 racionais que são utilizados como argumentos ($N1=3/5$, $N2=2/3$, $N3=1/5$, $N4=7/4$, $N5=1/2$, $N6=0/3$, $N7=5/6$ e $N8=7/0$), à medida que as expressões são calculadas. Os racionais também devem ser liberados à medida que não são mais utilizados.

Expressão 1: $X = (3/5 + 2/3) * (1/5 + 7/4)$

Expressão 2: $Y = (7/4 * 1/2) + (0/3 * 5/6)$

Expressão 3: $Z = 5/6 * (3/5 + 7/0)$ *Sugestão: repetir cálculo de Z com $N8=7/1$

3. Especificar e implementar de forma independente o TAD ponto, para representar pontos no espaço bidimensional (R^2). Este TAD deve permitir criar de um ponto, liberar a memória alocada para um ponto e calcular a distância entre dois pontos. O programa principal (aplicativo) deve ler as coordenadas de 2 pontos (digitadas pelo usuário) e imprimir na tela a distância entre estes pontos. Para isto, este aplicativo deverá criar os dois pontos, calcular e mostrar a distância entre eles e depois liberar a memória alocada para os pontos.
4. Acrescentar na TAD Racional (exerc. 2), a operação que simplifica um no. racional.