

Estruturas de Dados

PA05 - Tabela Hash Dinâmica

Professores: Wagner Meira Jr
Eder Figueiredo

Introdução

Nesta atividade, você irá implementar um TAD (Tipo Abstrato de Dados) que utiliza uma tabela hash dinâmica com rehashing linear. Essa abordagem é fundamental para compreender como estruturas de dados podem gerenciar coleções de informações de forma eficiente, permitindo operações rápidas como inserção, remoção e busca.

Objetivo da atividade

Você deverá implementar uma estrutura de conjuntos de strings utilizando uma tabela hash dinâmica. Seu TAD deverá oferecer as seguintes funcionalidades:

- **Inserção:** Adiciona um elemento ao conjunto, garantindo que não haja duplicatas.
- **Remoção:** Exclui um elemento do conjunto, caso ele exista.
- **Pertinência:** Verifica se um elemento pertence ao conjunto.
- **União:** Combina dois conjuntos, formando um novo que contém todos os elementos de ambos.
- **Interseção:** Retorna os elementos que são comuns aos dois conjuntos.
- **Diferença Simétrica:** Retorna os elementos que pertencem a apenas um dos conjuntos, descartando os que estão em ambos.

Nos arquivos disponíveis a função main já está implementada, você deve se preocupar apenas em implementar as funcionalidades listadas acima em seu TAD.

Revisando rehashing linear

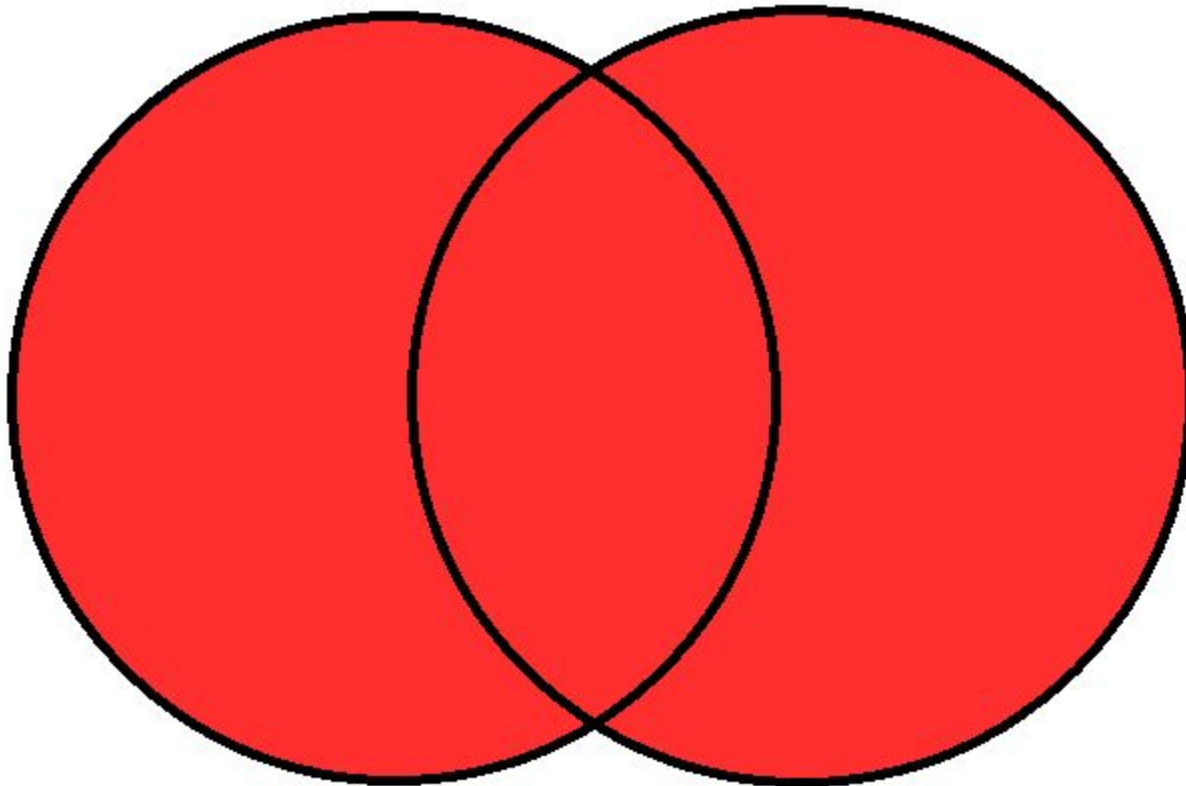
Quando a tabela atinge sua capacidade máxima, o rehashing linear é acionado. O procedimento é o seguinte:

1. Crie uma nova tabela, maior e vazia.
2. Percorra todos os elementos da tabela antiga e insira-os na nova tabela utilizando a função de hash atualizada.
3. Após transferir todos os elementos, libere a memória ocupada pela tabela antiga.

Este processo tem complexidade de tempo $O(M)$, onde M é o tamanho da tabela, e garante que a estrutura mantenha um desempenho eficiente mesmo com o aumento do número de elementos.

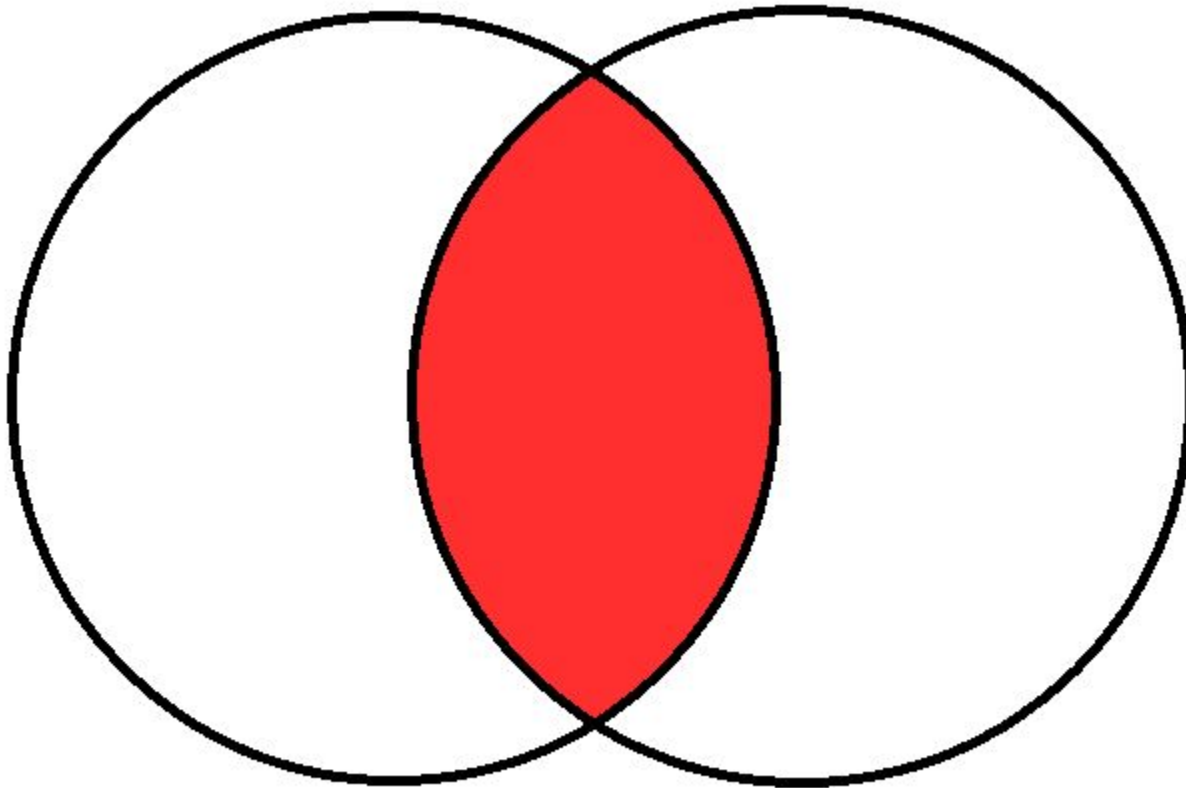
Revisão de conjuntos

Se A e B são conjuntos, a união de A com B é:



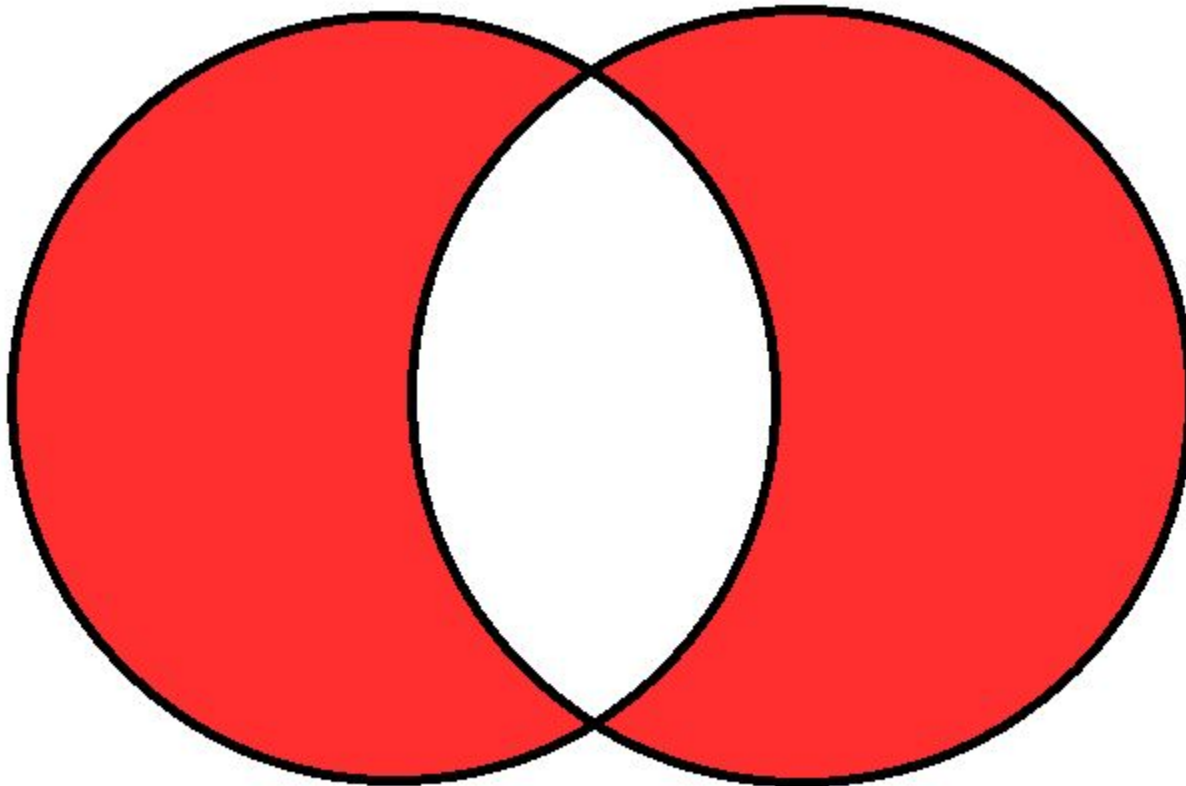
Revisão de conjuntos

Se A e B são conjuntos, a interseção de A com B é:



Revisão de conjuntos

Se A e B são conjuntos, a diferença simétrica de A com B é:



Roteiro da atividade

Observações sobre a implementação:

- Nos arquivos disponíveis a leitura da entrada já está implementada, você deve se concentrar apenas em implementar as operações da tabela hash e de conjuntos.
- Caso julgue necessário você pode incluir funções ou métodos novos no arquivo de cabeçalho.

Submissão

- A submissão será feita por **VPL**. Certifique-se de seguir as instruções do tutorial disponibilizado no moodle.
- O seu arquivo executável **DEVE** se chamar **pa5.out** e deve estar localizado na pasta **bin**.
- Seu código será compilado com o comando:
 make all
- Você **DEVE** utilizar a estrutura de projeto abaixo junto ao Makefile :
 - PA5
 - |- src
 - |- bin
 - |- obj
 - |- include
 - Makefile