

Universidade do Estado de Santa Catarina (UDESC)
Curso: Bacharelado em Ciência da Computação
Disciplina: Linguagem de Programação (LPG0001)
Prof. Rui Jorge Tramontin Jr.

Trabalho 1: Gerenciamento de Conjuntos

O objetivo do trabalho é implementar um programa que gerencie uma quantidade indeterminada de **conjuntos de valores inteiros**. O modelo consiste no uso de uma **matriz $M \times N$** , sendo M e N constantes.

Cada linha da matriz é usada para representar um conjunto de valores, sendo que o programa deve garantir que os limites da matriz não sejam ultrapassados. Portanto, não deve ser possível criar mais do que M conjuntos, e cada conjunto não pode ter mais do que N valores.

A matriz deve ser inicializada com zeros, indicando que está vazia. O zero, portanto, não é considerado como um valor válido.

Além da matriz, o programa deve utilizar um **contador** para registrar quantidade de conjuntos. Portanto, o contador também deve estar zerado ao início do programa.

O programa deve ficar em um ciclo de repetições, oferecendo um *menu de texto* com as seguintes opções ao usuário:

1. **Criar um novo conjunto vazio:** basta incrementar o contador, caso seja menor que M ;
2. **Inserir dados em um conjunto:** o usuário deve informar o índice i do conjunto em que deseja fazer a inserção ($0 \leq i < \text{contador}$); os demais requisitos devem ser respeitados:
 - a. O usuário pode inserir vários valores na sequência; o processo se encerra quando é digitado o valor zero ou quando se atingiu a quantidade N (número de colunas da matriz);
 - b. A inserção é feita sempre “à direita” na linha da matriz; um outro contador deve ser utilizado para indicar o próximo índice a ser inserido, sendo incrementado a cada inserção;
 - c. Como um conjunto deve possuir valores distintos, é preciso garantir que não há valores repetidos na entrada; portanto, deve-se verificar se o valor já foi inserido previamente; a função de *busca sequencial* implementada em aula pode ser usada aqui, facilitando o processo;
3. **Remover um conjunto:** o usuário deve informar o índice i do conjunto em que deseja fazer a inserção ($0 \leq i < \text{contador}$); a linha i deve ser zerada, e deve-se mover os conjuntos abaixo dele para reorganizar a matriz, decrementando-se o contador (pois há um conjunto a menos); por exemplo, tem-se 10 conjuntos e o usuário deseja remover o conjunto 6; neste caso, a linha 6 é zerada, os conjuntos (linhas da matriz) de 7 a 9 são movidos “para cima”, e o contador é decrementado para 9;

4. **Fazer a *união* entre dois conjuntos:** o usuário deve informar os índices dos conjuntos que fazem parte dessa operação; o conjunto resultante é adicionado à matriz de conjuntos como uma nova linha da matriz;
5. **Fazer a *intersecção* entre dois conjuntos:** idem à operação anterior;
6. **Mostrar um conjunto,** dado o seu índice;
7. **Mostrar todos os conjuntos;**
8. **Busca por um valor:** dado um valor, deve-se mostrar os índices dos conjuntos que contenham esse valor;
9. **Sair do programa.**

Requisitos do programa (e algumas dicas)

- Defina as constantes M e N para serem as dimensões da matriz; utilize as constantes para verificações e garantir que tais limites não sejam ultrapassados;
- A maioria das funcionalidades pode ser feita usando **funções**, o que facilita a implementação e valoriza a nota do trabalho!
- Caprichem nas mensagens para o usuário! Por exemplo, trate adequadamente quando não houver nenhum conjunto e o usuário escolher a opção para mostrar todos os conjuntos;
- Além disso, trate situações excepcionais, tais como quando o usuário escolher o índice de um conjunto que não existe.

Critérios de avaliação

- Execução correta e alinhamento com o que foi solicitado neste enunciado.

Informações importantes:

- **Trabalho individual ou em duplas;**
- **Entrega via Moodle.**