

Situação: O programa aborda a gestão de uma lista de contatos, que o usuário pode adicionar, remover, buscar e mostrar as informações dos contatos.

Problema: O programa aborda a necessidade de armazenar e gerenciar dados dos contatos de um jeito organizado e eficaz, para isso acontecer usei um array para armazenar os contatos e são fornecidos os métodos adicionar, remover, buscar e exibir na classe Vetor.

Alguns problemas que podem acontecer com o passar do tempo são: Capacidade máxima atingida, quando o usuário tentar adicionar um novo contato não será possível pois a máxima do vetor já estará atingida e vai aparecer uma mensagem que a capacidade máxima já foi atingida.

Remoção de contatos, o programa permite a remoção de contatos com base no nome fornecido pelo usuário, então se tiver dois nomes iguais o primeiro contato será apagado.

Buscar contatos, ela é feita com base no nome fornecido pelo usuário, então se tiver contatos com nomes parecidos o usuário não vai encontrar o contato que procura.

Exibir contatos, ela é feita de forma que seja apenas listado todos os contatos armazenados no vetor, isso pode ser ruim para o usuário pois não será muito prático.

Códigos do programa:

```
public class Contato {
    private String nome;
    private String telefone;

    public Contato(String nome, String telefone) {
        this.nome = nome;
        this.telefone = telefone;
    }

    public String getNome() {
        return nome;
    }

    public String getTelefone() {
        return telefone;
    }

    @Override
    public String toString() {
        return "Nome: " + nome + ", Telefone: " + telefone;
    }
}
```

```
public class Vetor {

    private Contato[] contatos;
    private int tamanho;

    public Vetor(int capacidade) {
```

```

        contatos = new Contato[capacidade];
        tamanho = 0;
    }

    public void adiciona(Contato contato) {
        if (tamanho < contatos.length) {
            contatos[tamanho] = contato;
            tamanho++;
        } else {
            System.out.println("Capacidade máxima atingida");
        }
    }

    public void remover(String nome) {
        int posicao = busca(nome);
        if (posicao != -1) {
            for (int i = posicao; i < tamanho - 1; i++) {
                contatos[i] = contatos[i + 1];
            }
            tamanho--;
            System.out.println("Contato removido com sucesso");
        } else {
            System.out.println("Contato não encontrado");
        }
    }

    private int busca(String nome) {
        for (int i = 0; i < tamanho; i++) {
            if (contatos[i].getNome().equals(nome)) {
                return i;
            }
        }
        return -1;
    }

    public void listaContatos() {
        if (tamanho == 0) {
            System.out.println("Nenhum contato encontrado");
        } else {
            System.out.println("Listando contatos:");
            for (int i = 0; i < tamanho; i++) {
                System.out.println(contatos[i]);
            }
        }
    }
}

```

```
import java.util.Scanner;
```

```

public class TesteVetor {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Vetor vetor = new Vetor(5);
    }
}

```

```

int opcao = 0;
while (opcao != 4) {
    System.out.println("1 - Adicionar contato");
    System.out.println("2 - Remover contato");
    System.out.println("3 - Listar contatos");
    System.out.println("4 - Sair");
    System.out.print("Digite a opção desejada: ");
    opcao = scanner.nextInt();
    scanner.nextLine();
    switch (opcao) {
        case 1:
            System.out.print("Digite o nome do contato: ");
            String nome = scanner.nextLine();
            System.out.print("Digite o telefone do contato: ");
            String telefone = scanner.nextLine();
            Contato contato = new Contato(nome, telefone);
            vetor.adiciona(contato);
            break;
        case 2:
            System.out.print("Digite o nome do contato a ser removido: ");
            nome = scanner.nextLine();
            vetor.remover(nome);
            break;
        case 3:
            vetor.listaContatos();
            break;
        case 4:
            System.out.println("Saindo...");
            break;
        default:
            System.out.println("Opção inválida");
    }
}
}
}

```