

Exercício: Sistema de Priorização de Contato para Ofertas de Energia Sustentável

O objetivo deste exercício é desenvolver um sistema para priorizar o contato de uma empresa de energia sustentável com seus clientes, oferecendo promoções e novas tecnologias de consumo de energia limpa. O sistema deve organizar e filtrar os clientes com base em seu consumo de energia e em sua elegibilidade para ofertas especiais.

Descrição do Sistema

1. **Cadastro de Clientes:** Cada cliente da empresa de energia será cadastrado no sistema com os seguintes dados:
 - Nome
 - CPF
 - Tipo de energia consumida (por exemplo: solar, eólica, elétrica)
 - Consumo Mensal de Energia (em kWh)
 - Apto para Oferta (Inicialmente, true)

O cadastro dos clientes será realizado em uma **árvore AVL**, onde o critério de organização dos registros será o **CPF** de cada cliente.

2. **Oferecimento de Energia Sustentável:** Periodicamente, a empresa decide lançar ofertas para promover o uso de fontes renováveis de energia, como a instalação de painéis solares. O gerente informa um valor mínimo de consumo para qualificar o cliente para a oferta. O programa percorre a árvore AVL de cadastro de clientes e gera uma nova árvore AVL, chamada **Árvore de Oferta**, com os clientes que atendem ao critério de consumo.

A nova árvore será organizada pela quantidade de energia consumida, de forma **decrescente**, dando prioridade aos clientes com maior consumo de energia, para que possam ser priorizados na oferta de soluções sustentáveis.

3. **Simulação de Contato com os Clientes:** A **Árvore de Oferta** gerada será utilizada para simular o contato com os clientes. O programa realiza o seguinte processo:
 - Se o cliente aceitar a oferta de energia sustentável (por exemplo, a instalação de painéis solares), o atributo **Apto para Oferta** do cliente é alterado para false na árvore de cadastro.
 - Se o cliente não aceitar a oferta, seu registro permanece inalterado na árvore de cadastro.

4. **Consulta ao Cadastro de Clientes:** O sistema deve permitir consultas sobre o cadastro de clientes. O gerente pode acessar as seguintes funcionalidades:
 - **Consulta de cliente por CPF:** Exibe todos os dados do cliente, incluindo tipo de energia consumida e consumo mensal.
 - **Somatório do Consumo de Energia:** Calcula o total de energia consumida por todos os clientes cadastrados.
 - **Quantidade de clientes com consumo acima de um valor mínimo:** Conta quantos clientes têm consumo de energia superior a um valor especificado.
5. **Encerramento do Programa:** Antes de encerrar o sistema, ele deve exibir todos os clientes que não aceitaram a oferta ou que não atendem aos critérios de elegibilidade para receber as ofertas sustentáveis.

Funcionalidades

O sistema em Java deve ter um menu principal com as seguintes opções:

1. **Inscrição de Cliente:** O gerente preenche as informações do cliente e o insere na árvore AVL de cadastro.
2. **Oferta de Energia Sustentável:** O gerente informa o valor mínimo de consumo de energia para qualificar o cliente para a oferta. A árvore AVL de cadastro é percorrida, e uma nova árvore AVL é criada com os clientes aptos.
3. **Consulta ao Cadastro:** O gerente pode acessar um submenu para realizar consultas sobre o cadastro de clientes:
 - Consulta por CPF.
 - Somatório do consumo de energia de todos os clientes.
 - Quantidade de clientes com consumo de energia acima de um valor mínimo.
4. **Encerrar o Programa:** Antes de encerrar, o sistema deve exibir todos os clientes que não aceitaram a oferta.

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

```
public class EnergiaSustentavel {
```

```
public static void main(String[] args) {

    Scanner le = new Scanner(System.in);

    AVLCliente cadastro = new AVLCliente();

    AVLCliente oferta = new AVLCliente();

    int opcao, op;

    String nome, tipoEnergia, cpf;

    double consumoMensal;

    do {

        System.out.println("0 - Encerrar o programa");

        System.out.println("1 - Inscrição de um cliente");

        System.out.println("2 - Oferta de energia sustentável");

        System.out.println("3 - Entrar no Submenu");

        opcao = le.nextInt();

        switch (opcao) {

            case 0:

                System.out.println("\n\nClientes que não aceitaram ou não estavam aptos para a oferta:");

                cadastro.imprimirClientesNaoAceitaramOferta();

                break;

            case 1:

                System.out.print("Digite nome: ");

                le.nextLine(); // Consumir a linha de quebra

                nome = le.nextLine();

                System.out.print("Digite CPF: ");

                cpf = le.next();

                System.out.print("Tipo de energia consumida (solar, eólica, elétrica): ");

                tipoEnergia = le.next();
```

```

        System.out.print("Informe consumo mensal de energia (kWh): ");
        consumoMensal = le.nextDouble();

        Cliente cliente = new Cliente(nome, cpf, tipoEnergia, consumoMensal,
true);

        cadastro.root = cadastro.inserir(cadastro.root, cliente);

        break;

    case 2:

        System.out.print("Qual o valor mínimo de consumo de energia (kWh)
para a oferta? ");

        consumoMensal = le.nextDouble();

        oferta = cadastro.gerarOferta(consumoMensal);

        oferta.gerarFilaClientes();

        break;

    case 3:

        do {

            System.out.println("\t1) Consulta cliente buscando pelo CPF");

            System.out.println("\t2) Apresenta o total de consumo de energia de
todos os clientes");

            System.out.println("\t3) Apresenta a quantidade de clientes com
consumo acima de um valor");

            System.out.println("\t4) Voltar menu principal");

            op = le.nextInt();

            switch (op) {

                case 1:

                    System.out.print("Informe CPF para consulta: ");

                    cpf = le.next();

                    cadastro.consultarPorCpf(cpf);

```

```
        break;
    case 2:
        cadastro.somarConsumoTodosClientes();
        break;
    case 3:
        System.out.print("Qual valor mínimo de consumo de energia para
consulta? ");

        double minimo = le.nextDouble();
        cadastro.contarClientesAcimaDeMinimo(minimo);
        break;
    case 4:
        break;
    default:
        System.out.println("Opção inválida");
    }
} while (op != 4);
break;
default:
    System.out.println("Opção inválida");
}
} while (opcao != 0);
le.close();
}
}
```