

ADO2 ED

Main.java

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Fila fila = new Fila(50);
        int opcao;

        do {
            System.out.println("1 - Solicitar Senha (Normal)");
            System.out.println("2 - Solicitar Senha (Prioridade)");
            System.out.println("3 - Listar as senhas");
            System.out.println("4 - Espiar");
            System.out.println("5 - Chamar próxima pessoa");
            System.out.println("0 - Sair");
            System.out.println("Escolha uma opção: ");
            opcao = sc.nextInt();
            sc.nextLine();
            switch (opcao) {
                case 1: {
                    System.out.println("Digite o nome: ");
                    String nome = sc.nextLine();
                    Pessoa p = new Pessoa(nome, false);
                    fila.enqueue(p);
                    System.out.println("Senha normal adicionada: " + p);
                    break;
                }
                case 2: {
                    System.out.println("Digite o nome: ");
                    String nome = sc.nextLine();
                    Pessoa p = new Pessoa(nome, true);
                    fila.enqueue(p);
                    System.out.println("Senha prioritária adicionada: " + p);
                    break;
                }
            }
        } while (opcao != 0);
    }
}
```

```

    fila.enqueue(p);
    System.out.println("Senha prioritária adicionada: " + p);
    break;
}
case 3: {
    System.out.println("Fila atual: " + fila);
    break;
}
case 4: {
    Pessoa frente = fila.front();
    if (frente == null) {
        System.out.println("Fila está vazia.");
    } else {
        System.out.println("Próximo a ser chamado: " + frente);
    }
    break;
}
case 5: {
    Pessoa chamado = fila.call();
    if (chamado != null) {
        System.out.println("Chamando: " + chamado);
    }
    break;
}
case 0: {
    System.out.println("Saindo");
    break;
}
default:
    System.out.println("Opção inválida");
}
} while (opcao != 0);
}
}

```

Fila.java

```

public class Fila {

    private Pessoa[] v;
    private int tamanho;
    private int chamadaPrioritarios;

    public Fila(int tamanho) {
        this.v = new Pessoa[tamanho];
        this.tamanho = 0;
    }

    public void enfileirar(Pessoa p) {
        if (p.isPrioritario()) {
            int posicao = buscarPosicaoNormal();
            if (posicao == -1) {
                v[tamanho] = p;
            } else {
                for (int i = tamanho; i > posicao; i--) {
                    v[i] = v[i - 1];
                }
                v[posicao] = p;
            }
        } else {
            v[tamanho] = p;
        }
        tamanho++;
    }

    public void enfileirarPrioridade(Pessoa p) {
        if (p.isPrioritario()) {
            int posicao = buscarPosicaoNormal();
            if (posicao == -1) {
                v[tamanho] = p;
            } else {
                for (int i = tamanho; i > posicao; i--) {
                    v[i] = v[i - 1];
                }
            }
        }
    }
}

```

```

        v[posicao] = p;
    }
} else {
    v[tamanho] = p;
}
tamanho++;
}

@Override

public String toString() {
    StringBuilder s = new StringBuilder("[");
    for (int i = 0; i < tamanho; i++) {
        s.append(v[i]);
        if (i < tamanho - 1) {
            s.append(", ");
        }
    }
    s.append("]");
    return s.toString();
}

public Pessoa verFrente() {
    if (tamanho == 0) {
        System.out.println("A pilha está vazia");
        return null;
    }

    return v[0];
}

public Pessoa chamarFila() {
    Pessoa pessoaChamada = null;
    int posicao = 0;
    if (tamanho == 0) {
        System.out.println("A fila está vazia");
        return null;
    }
}

```

```

if (chamadaPrioritarios < 3) {
    posicao = buscarPosicaoPrioridade();
    if (posicao == -1) {
        posicao = 0;
        chamadaPrioritarios = 0;
    }
    pessoaChamada = v[posicao];
    for (int i = posicao; i < tamanho - 1; i++) {
        v[i] = v[i + 1];
    }
    tamanho--;
    if (pessoaChamada.isPrioritario()) {
        chamadaPrioritarios++;
    }
    return pessoaChamada;
} else {
    posicao = buscarPosicaoNormal();
    pessoaChamada = v[posicao];
    for (int j = posicao; j < tamanho - 1; j++) {
        v[j] = v[j + 1];
    }
    if (chamadaPrioritarios == 3) {
        chamadaPrioritarios = 0;
    }
    tamanho--;
}
return pessoaChamada;
}

public int buscarPosicaoPrioridade() {
    for (int i = 0; i < tamanho; i++) {
        if (v[i].isPrioritario()) {
            return i;
        }
    }
    return -1;
}

```

```

public int buscarPosicaoNormal() {
    for (int i = 0; i < tamanho; i++) {
        if (!v[i].isPrioritario()) {
            return i;
        }
    }
    return -1;
}

public int retornarTamanho() {
    return tamanho;
}

}

```

Pessoa.java

```

public class Pessoa {
    private String nome;
    private boolean prioritario;

    public Pessoa(String nome, boolean prioritario){
        this.nome = nome;
        this.prioritario = prioritario;
    }

    public String getNome() {
        return nome;
    }
    public void setNome(String nome) {
        this.nome = nome;
    }

    public boolean getPrioritario(){
        return prioritario;
    }
}

```

```
}

public void setPrioritario(boolean prioritario) {
    this.prioritario = prioritario;
}

public boolean isPrioritario(){
    return getPrioritario();
}

@Override
public String toString() {
    return nome + (prioritario ? " (Prioritário)" : " (Normal)");
}
```