

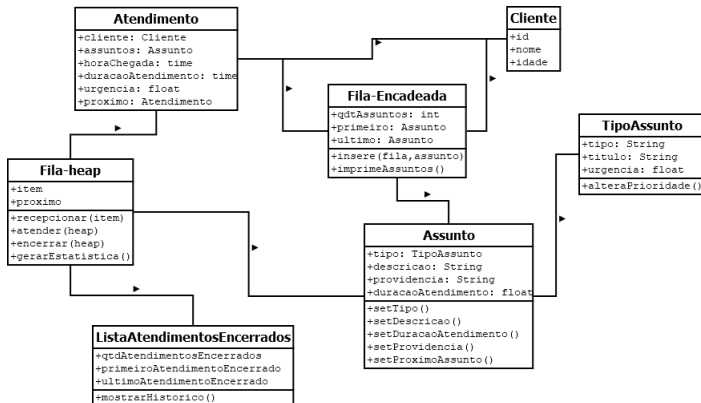
Trabalho de Implementação Estrutura de Dado

Luiz Gustavo Dias

IC/UFF

25/06/2018

Visão Geral



Sequência de Comandos

- `recepcionar()`; Adiciona o cliente, suas demandas, a lista de assuntos padrão, a hora de chegada e hora de atendimento do cliente.

```
Atendimento novo = new Atendimento(cliente, assuntosCliente, horaChegada, horaAtendimento, null);
System.out.println("\nComando: ***Recepcionar Cliente***\n");
if (tamanhoFila == atendimentos.length-1) {
    System.out.println("A fila de clientes está cheia! Encerre atendimentos para continuar!");
}else {
    tamanhoFila++;
    novo.urgencia = calculaPriridade(novo);
    atendimentos[tamanhoFila] = novo;
    imprimeAtendimentos(atendimentos[tamanhoFila]);
}
```

Sequência de Comandos

- ▶ atender(); Seleciona o cliente com maior prioridade e coloca-o no atendimento; É nessa função que é realizado o heap, pois a prioridade de algum assunto pode ter sido alterada em algum momento, o que pode alterar a prioridade dos clientes recepcionados;

```
int filho = 2*i+1;
if (filho <= tamanhoFila ) {
    if (filho <= tamanhoFila - 1) {

        if (atendimentos[i].urgencia > atendimentos[filho].urgencia) {
            filho = filho + 1;
        }

        if (atendimentos[i].urgencia < atendimentos[filho].urgencia) {
            //System.out.println("i: "+atendimentos[i].urgencia+"é menor que '
            Atendimento aux = atendimentos[i];
            atendimentos[i] = atendimentos[filho];
            atendimentos[filho] = aux;
            descer(filho);
        }
    }
}
```

Sequência de Comandos

- ▶ `encerrar()`; Envia os dados do cliente atendido para a função `incrementaEstatistica()`, que tria os dados do atendimento e os organiza em forma de lista;

```
incrementaEstatistica(atendimentos[0], listaPadrao);
```

Sequência de Comandos

- ▶ `gerarEstatistica()`: mostra estatísticas de atendimentos atendidos até o momento;