

## EXPLICAÇÃO DA SOLUÇÃO

### MENU INICIAL

#### Implementação

- Foi utilizado o Scanner para a coleta de dados fornecidos pelo usuário;
- O comando “System.out.print” mostra as opções possível na tela do usuário;
- O comando “switch” permite que o usuário opte por uma das opções em tela (números de 1 a 6);
- O comando “do-while” faz com que as opções implementadas internamente sejam repedidas, isso faz com que o menu reapareça após cada resultado obtido ou até que o usuário opte pela opção “Terminar o programa”;
- Dentro de cada “case” no comando switch nós encontramos as chamadas dos métodos para cada ação a ser executada.

#### Capturas de Tela

➔ Código de execução:

```
10 public static void main(String[] args) {
11
12     Scanner sc = new Scanner(System.in);
13
14     String aux;
15     do {
16         System.out.print("\n ----- SEJA BEM-VINDO(A) ----- "
17             + "\n\n  1. Consultar média de idade dos pacientes\n"
18             + "  2. Consultar internações por ano\n"
19             + "  3. Consultar hospitais\n"
20             + "  4. Calcular tempo de internação\n"
21             + "  5. Determinar tempos de espera na fila\n"
22             + "  6. Terminar o programa\n"
23             + "\nDigite a opção desejada: ");
24
25
26         aux = sc.nextLine();
27         System.out.print("\n ----- \n");
28         switch (aux) {
29             case "1":
30                 Scanner le1 = new Scanner(System.in);
31                 System.out.print("\nFavor informar o nome do município residencial: ");
32                 String x1= le1.nextLine();
33                 opcoesMenu y1 = new opcoesMenu();
34                 y1.opcao1(x1);
35                 break;
36             case "2":
37                 Scanner le2 = new Scanner(System.in);
38                 System.out.print("\nFavor informar o nome do município residencial: ");
39                 String x2= le2.nextLine();
40                 opcoesMenu y2 = new opcoesMenu();
41                 y2.opcao2(x2);
42                 break;
43             case "3":
44                 Scanner le3 = new Scanner(System.in);
45                 System.out.print("\nFavor informar o nome da instituição de saúde executante: ");
46                 String x3= le3.nextLine();
47                 opcoesMenu y3 = new opcoesMenu();
48                 y3.opcao3(x3);
49                 break;
50             case "4":
51                 Scanner le4 = new Scanner(System.in);
52                 System.out.print("\nFavor informar o nome da instituição de saúde solicitante: ");
53                 String x4= le4.nextLine();
54                 opcoesMenu y4 = new opcoesMenu();
55                 y4.opcao4(x4);
56                 break;
57             case "5":
58                 opcoesMenu y5 = new opcoesMenu();
59                 y5.opcao5();
60                 break;
61             case "6":
62                 opcoesMenu y6 = new opcoesMenu();
63                 y6.opcao6();
64                 break;
65             default:
66                 System.out.print("\nOps... opção inválida. Tente novamente ! \n");
67         }
68     } while (aux != "6");
69 }
70 }
```

➔ Resultados:

```
----- SEJA BEM-VINDO(A) -----  
  
    1. Consultar média de idade dos pacientes  
    2. Consultar internações por ano  
    3. Consultar hospitais  
    4. Calcular tempo de internação  
    5. Determinar tempos de espera na fila  
    6. Terminar o programa  
  
Digite a opção desejada: |
```

### Testes

Os testes da implementação foram manuais (por tentativas de fornecimento de dados inválidos).

```
----- SEJA BEM-VINDO(A) -----  
  
    1. Consultar média de idade dos pacientes  
    2. Consultar internações por ano  
    3. Consultar hospitais  
    4. Calcular tempo de internação  
    5. Determinar tempos de espera na fila  
    6. Terminar o programa  
  
Digite a opção desejada: gdsfkajhds  
|  
-----  
Ops... opção inválida. Tente novamente !  
  
----- SEJA BEM-VINDO(A) -----  
  
    1. Consultar média de idade dos pacientes  
    2. Consultar internações por ano  
    3. Consultar hospitais  
    4. Calcular tempo de internação  
    5. Determinar tempos de espera na fila  
    6. Terminar o programa  
  
Digite a opção desejada:
```

As capturas da utilização do menu constam na documentação abaixo em cada uma das respectivas funcionalidades exigidas no exercício.

## **CLASSE INTERNAÇÕES E LEITURA DO ARQUIVO**

### Implementação

- Nesta classe “internacoes” nós temos o método que irá ler o arquivo, os métodos construtores (permite a criação dos objetos), os métodos getters e setters (permitem acesso e edição de cada variável do objeto), e o método toString() (permite mostrar em tela as informações do objeto);

- Um objeto ArrayList de interações é criado para que possamos receber os dados da planilha a ser lida;
- O método “LerCSV” recebe o diretório onde o arquivo .csv se encontra na máquina e devolver um arraylist de “internacoes”, ele utiliza o Buffered Reader para ler o arquivo;
- No método são utilizados “try-catch” para identificação de possíveis erros, estes erros, se ocorrerem, são sinalizados no console;
- O comando “while” é utilizado para que enquanto cada linha do arquivo for diferente de “null” o laço prossiga, neste laço é implementado um primeiro “if” que considera o cabeçalho existe ou não (“eCabecalho” vem como “true” já que o nosso documento tem cabeçalho)
- Na sequência, dentro do laço, criamos um Array de Strings onde as informações de cada objeto serão armazenadas, dentro do comando split() informamos que serão 19 colunas e que o divisor de cada coluna, por ser um arquivo .csv será o caracter “;”;
- Em seguida, ainda dentro do laço, nós tratamos as colunas que potencialmente poderão estar com células vazias evitando erros deste tipo (utilizamos mais um “if” e também o método “isEmpty()”);
- Declaramos as variáveis, definimos os seus tipos e formatos (utilização do método SimpleDateFormat()), efetuamos as devidas conversões (utilização do método parse());
- Criamos um novo objeto “interna” do tipo “internacoes” que receberá as informações de uma linha do arquivo, por fim adicionamos o objeto “interna” no ArrayList de objetos tipo “internacoes” (utilizando o método add());
- Por fim, as ações dentro do comando “while” se repetem até incluirmos todos as linhas/objetos “interna” no ArrayList “internacoes” que será retornado pelo método lerCSV.

### Capturas de Tela

➔ Código de execução:

```

1 package entities;
2
3 import java.io.BufferedReader;
4
11 public class internacoes{
12
13     private Double idade;
14     private Double horasNaFila;
15     private Date dataExtracao;
16     private Date dataSolicitacao;
17     private Date dataAutorizacao;
18     private Date dataInternacao;
19     private Date dataAlta;
20     private String idUsuario, situacao, centralRegOrigem, sexo, municipioResidencia, solicitante,
21         municipioSolicitante, codCid, carater, tipoInternacao, tipoLeito, executante;
22
23     public internacoes(){
24     }
25
26
27     public static ArrayList<internacoes> lerCSV(String caminhoDoArquivo){
28
29         ArrayList<internacoes> internacoes = new ArrayList<>();
30
31         try (BufferedReader br = new BufferedReader(new FileReader(caminhoDoArquivo))) {
32             String linha;
33             boolean eCabecalho = true;
34             while ((linha = br.readLine()) != null) {
35                 if (eCabecalho || linha.trim().isEmpty()) {
36                     eCabecalho = false;
37                     continue;
38                 }
39                 String[] columnas = linha.split(";", 19);
40                 if (columnas.length != 19) {
41                     System.out.println("O CSV tem mais ou menos de 19 columnas!");
42                     continue;
43                 }
44                 try {
45                     try {
46                         Double idade=0.0;
47                         if (!columnas[6].isEmpty()) {
48                             idade = Double.parseDouble(columnas[6]);
49                         }
50
51                         Double horasNaFila = Double.parseDouble(columnas[18]);
52                         Date dataExtracao = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(columnas[0]);
53                         Date dataAutorizacao = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(columnas[14]);
54                         Date dataInternacao = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(columnas[15]);
55
56                         String aux = "2000-00-00 00:00:00.000";
57                         Date dataAlta = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux);
58                         if (!columnas[16].isEmpty()) {
59                             dataAlta = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(columnas[16]);
60                         }
61
62                         Date dataSolicitacao = new SimpleDateFormat("yyyy-MM-dd").parse(columnas[4]);
63
64                         String idUsuario = columnas[1], situacao = columnas[2], centralRegOrigem = columnas[3], sexo = columnas[5],
65                             municipioResidencia = columnas[7], solicitante = columnas[8], municipioSolicitante = columnas[9],
66                             codCid = columnas[10], carater = columnas[11], tipoInternacao = columnas[12],
67                             tipoLeito = columnas[13], executante = columnas[17];
68
69                         internacoes interna = new internacoes();
70
71                         interna.setDataExtracao(dataExtracao);
72                         interna.setIdUsuario(idUsuario);
73                         interna.setSituacao(situacao);
74                         interna.setCentralRegOrigem(centralRegOrigem);
75                         interna.setDataSolicitacao(dataSolicitacao);
76                         interna.setSexo(sexo);
77                         interna.setIdade(idade);
78                         interna.setMunicipioResidencia(municipioResidencia);
79                         interna.setSolicitante(solicitante);
80                         interna.setMunicipioSolicitante(municipioSolicitante);
81                         interna.setCodCid(codCid);
82                         interna.setCarater(carater);
83                         interna.setTipoInternacao(tipoInternacao);
84                         interna.setTipoLeito(tipoLeito);
85                         interna.setDataAutorizacao(dataAutorizacao);
86                         interna.setDataInternacao(dataInternacao);
87                         interna.setDataAlta(dataAlta);
88                         interna.setExecutante(executante);
89                         interna.setHorasNaFila(horasNaFila);
90
91                         internacoes.add(interna);
92
93                     } catch (ParseException e){
94                         System.out.println("ParseException occurred: "+e.getMessage());
95                     }
96                 } catch (Exception ex) {
97                     System.out.println("Deu algum problema!\n " + ex);
98                 }
99             }
100         } catch (FileNotFoundException e) {
101             e.printStackTrace();
102         } catch (IOException e) {
103             e.printStackTrace();
104         }
105
106         return internacoes;
107     }

```

```

108
109• public Double getIdade() {
110    return idade;
111}
112
113• public void setIdade(Double idade) {
114    this.idade = idade;
115}
116
117• public Double getHorasNaFila() {
118    return horasNaFila;
119}
120
121• public void setHorasNaFila(Double horasNaFila) {
122    this.horasNaFila = horasNaFila;
123}
124
125• public Date getDataExtracao() {
126    return dataExtracao;
127}
128
129• public void setDataExtracao(Date dataExtracao) {
130    this.dataExtracao = dataExtracao;
131}
132
133• public Date getDataSolicitacao() {
134    return dataSolicitacao;
135}
136
137• public void setDataSolicitacao(Date dataSolicitacao) {
138    this.dataSolicitacao = dataSolicitacao;
139}
140
141• public Date getDataAutorizacao() {
142    return dataAutorizacao;
143}
144
145• public void setDataAutorizacao(Date dataAutorizacao) {
146    this.dataAutorizacao = dataAutorizacao;
147}
148
149• public Date getDataInternacao() {
150    return dataInternacao;
151}
152
153• public void setDataInternacao(Date dataInternacao) {
154    this.dataInternacao = dataInternacao;
155}
156
157• public Date getDataAlta() {
158    return dataAlta;
159}
160
161• public void setDataAlta(Date dataAlta) {
162    this.dataAlta = dataAlta;
163}
164
165• public String getIdUsuario() {
166    return idUsuario;
167}

```

```

168
169• public void setIdUsuario(String idUsuario) {
170    this.idUsuario = idUsuario;
171}
172
173• public String getSituacao() {
174    return situacao;
175}
176
177• public void setSituacao(String situacao) {
178    this.situacao = situacao;
179}
180
181• public String getCentralRegOrigem() {
182    return centralRegOrigem;
183}
184
185• public void setCentralRegOrigem(String centralRegOrigem) {
186    this.centralRegOrigem = centralRegOrigem;
187}
188
189• public String getSexo() {
190    return sexo;
191}
192
193• public void setSexo(String sexo) {
194    this.sexo = sexo;
195}
196
197• public String getMunicipioResidencia() {
198    return municipioResidencia;
199}
200
201• public void setMunicipioResidencia(String municipioResidencia) {
202    this.municipioResidencia = municipioResidencia;
203}
204
205• public String getSolicitante() {
206    return solicitante;
207}
208
209• public void setSolicitante(String solicitante) {
210    this.solicitante = solicitante;
211}
212
213• public String getMunicipioSolicitante() {
214    return municipioSolicitante;
215}
216
217• public void setMunicipioSolicitante(String municipioSolicitante) {
218    this.municipioSolicitante = municipioSolicitante;
219}
220
221• public String getCodCid() {
222    return codCid;
223}
224
225• public void setCodCid(String codCid) {
226    this.codCid = codCid;
227}

```

```

228
229• public String getCarater() {
230    return carater;
231}
232
233• public void setCarater(String carater) {
234    this.carater = carater;
235}
236
237• public String getTipoInternacao() {
238    return tipoInternacao;
239}
240
241• public void setTipoInternacao(String tipoInternacao) {
242    this.tipoInternacao = tipoInternacao;
243}
244
245• public String getTipoLeito() {
246    return tipoLeito;
247}
248
249• public void setTipoLeito(String tipoLeito) {
250    this.tipoLeito = tipoLeito;
251}
252
253• public String getExecutante() {
254    return executante;
255}
256
257• public void setExecutante(String executante) {
258    this.executante = executante;
259}
260
261• @Override
262• public String toString() {
263    return "internacoes [idade=" + idade + ", horasNaFila=" + horasNaFila + ", dataExtracao=" + dataExtracao
264    + ", dataSolicitacao=" + dataSolicitacao + ", dataAutorizacao=" + dataAutorizacao + ", dataInternacao="
265    + dataInternacao + ", dataAlta=" + dataAlta + ", idUsuario=" + idUsuario + ", situacao=" + situacao
266    + ", centralRegOrigem=" + centralRegOrigem + ", sexo=" + sexo + ", municipioResidencia="
267    + municipioResidencia + ", solicitante=" + solicitante + ", municipioSolicitante="
268    + municipioSolicitante + ", codCid=" + codCid + ", carater=" + carater + ", tipoInternacao="
269    + tipoInternacao + ", tipoLeito=" + tipoLeito + ", executante=" + executante + "]\n";
270}
271}
272

```

### ➔ Resultados:

- Esta classe “internacoes”, por fim, acaba por permitir a criação de objetos do tipo “internacoes”, o método LerSCV() capta as informações do arquivo .csv e transfere linha por linha para o ArrayList de objetos que será retornado;
- Além disso, os métodos getters e setters agora permitem que acessemos as variáveis de cada objeto, agora podendo consultá-las ou editá-las conforme a necessidade.

### Testes

- Durante a construção do código foram feitos diversos testes manuais, durante o processo, eram adicionados o comando de impressão em tela (System.out.println) para ter certeza de que o arquivo estava sendo reconhecido/lido adequadamente.
- Em todas os resultados, uma mensagem é apresentada instruindo o usuário a tentar novamente evitando caracteres especiais ou acentuação.

```
----- SEJA BEM-VINDO(A) -----
1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 1

-----

Favor informar o nome do município residencial: gdfsgfhs

-> O número total de pacientes do município: 0

-> A média de idade dos pacientes separados por gênero:

    MASCULINO: 0
    FEMININO: 0

-> A média de idade de todos os pacientes : 0

* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu
e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).

----- SEJA BEM-VINDO(A) -----
1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada:
```

## **FUNCIONALIDADE 1**

### Implementação

- Uma nova classe “opcoes” foi criada para armazenar os métodos que executariam cada uma das opções do menu inicial, foram então criados os métodos “opcoes1(), opcoes2(), opcoes3(), opcoes4(), opcoes5(), opcoes6()”;
- Iniciamos com a criação de uma cópia da listagem contida no arquivo .csv, é aqui que informamos o caminho do arquivo a ser copiado (informado por parâmetro ao chamarmos o método LerCSV(\*caminhodoarquivo\*));

- No método `opcao1()`, que se refere a esta primeira funcionalidade, o usuário fornece um parâmetro do tipo `String`;

- Implementamos o comando “for” para percorrer a lista, dentro do “for” adicionamos “ifs-else” que só serão acessados se o município informado pelo usuário for o mesmo que constar na linha/objeto, se atendida esta condição, então se verifica o gênero/sexo nesta mesma posição da lista (posição “i”), obedecendo as condições as variáveis auxiliares, de contagem e de soma são alteradas para posteriormente serem utilizadas para cálculo da média;

- Após o cálculo das médias, as informações são mostradas ao usuário em tela (para tirarmos os dígitos após o ponto, foi feito um “cast” para alterarmos as variáveis de “Double” para “int”).

### Capturas de Tela

Código de execução:

```

1 package menu;
2
3 import java.text.DecimalFormat;
4
5
6
7
8
9
10
11 public class opcoesMenu {
12
13     ArrayList<internacoes> lista1 = internacoes.lerCSV("c:\\teste\\gerint_solicitacoes_mod.csv");
14
15     public void opcao1(String nomeMunicipio) {
16
17         String nm = nomeMunicipio;
18         double totalPacientes=0.0;
19         double mediaIdadePorGeneroM=0.0;
20         double mediaIdadePorGeneroF=0.0;
21         double mediaIdadeGeral=0.0;
22
23         double contM=0.0,contF=0.0,contNI=0.0;
24         double somaM=0.0,somaF=0.0,somaNI=0.0;
25         double idadeAux=0.0;
26
27         for(int i=0;i<lista1.size();i++) {
28
29             if(nm.equalsIgnoreCase(lista1.get(i).getMunicipioResidencia())==true){
30                 totalPacientes++;
31
32                 if (lista1.get(i).getSexo().equalsIgnoreCase("MASCULINO")==true) {
33                     idadeAux = lista1.get(i).getIdade();
34                     contM++;
35                     somaM+=idadeAux;
36                 } else if (lista1.get(i).getSexo().equalsIgnoreCase("FEMININO")==true) {
37                     idadeAux = lista1.get(i).getIdade();
38                     contF++;
39                     somaF+=idadeAux;
40                 }
41             }
42         }
43         if(somaM!=0)
44             mediaIdadePorGeneroM=somaM/contM;
45
46         if(somaF!=0)
47             mediaIdadePorGeneroF=somaF/contF;
48
49         mediaIdadeGeral=(somaM+somaF+somaNI)/(contM+contF+contNI);
50
51         System.out.println("\n -> O número total de pacientes do município: "+(int)totalPacientes);
52         System.out.println("\n -> A média de idade dos pacientes separados por gênero:\n\n  MASCULINO: "+(int)mediaIdadePorGeneroM
53             +"\n  FEMININO: "+(int)mediaIdadePorGeneroF);
54         System.out.println("\n -> A média de idade de todos os pacientes : "+(int)mediaIdadeGeral);
55         System.out.print("\n\n* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu "
56             + "\n e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).\n");
57     }
58 }

```

## ➔ Resultados

```
----- SEJA BEM-VINDO(A) -----  
  
    1. Consultar média de idade dos pacientes  
    2. Consultar internações por ano  
    3. Consultar hospitais  
    4. Calcular tempo de internação  
    5. Determinar tempos de espera na fila  
    6. Terminar o programa  
  
Digite a opção desejada: 1  
|  
-----  
  
Favor informar o nome do município residencial:
```

```
----- SEJA BEM-VINDO(A) -----  
  
    1. Consultar média de idade dos pacientes  
    2. Consultar internações por ano  
    3. Consultar hospitais  
    4. Calcular tempo de internação  
    5. Determinar tempos de espera na fila  
    6. Terminar o programa  
  
Digite a opção desejada: 1  
|  
-----  
  
Favor informar o nome do município residencial: candido godoi  
|  
-> O número total de pacientes do município: 7  
  
-> A média de idade dos pacientes separados por gênero:  
  
    MASCULINO: 62  
    FEMININO: 64  
  
-> A média de idade de todos os pacientes : 63  
  
* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu  
  e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).  
  
----- SEJA BEM-VINDO(A) -----  
  
    1. Consultar média de idade dos pacientes  
    2. Consultar internações por ano  
    3. Consultar hospitais  
    4. Calcular tempo de internação  
    5. Determinar tempos de espera na fila  
    6. Terminar o programa  
  
Digite a opção desejada:
```



## Testes

Foram implementados testes manuais, após verificar que não o programa não estava reconhecendo caracteres minúsculos foi implementado o método equalsIgnoreCase() nos “ifs” para verificar se as Strings era iguais e também ignorar os maiúsculos e minúsculos.

## **FUNCIONALIDADE 2**

### Implementação

- No método opcao2(), o usuário fornece um parâmetro “nome de cidade” do tipo String;
- É criado um novo ArrayList “lista2” para receber os valores solicitados a serem impressos em tela;
- Também são criadas variáveis tipo “Date” que receberam valores no formato de hora utilizado na planilha original. Essas datas criadas correspondem às datas entre as quais queremos obter os resultados solicitados, ou seja, elas irão auxiliar a obter os quantitativos para cada ano de 2018,2019,2020 e 2021;
- É implementado o comando “try-catch” para identificar possíveis erros de conversão de variáveis;
- Seguindo o exemplo da funcionalidade anterior, percorremos a lista, e caso os valores na planilha atendam ao valor digitado pelo usuário, o programa utilizará os métodos after(\*dataInicioDoPeríodo\*) e before(\*dataFimDoPeríodo\*) nos seus “ifs” correspondentes para acessar o método add(), adicionando os objetos/linhas ao novo ArrayList a ser exibido, e acessar também as variáveis de contagem ();
- Por fim, é impresso em tela, através de um novo “for”, o ArrayList “internacoes” abastecido com os objetos/linhas que obedeceram o “município” informado pelo usuário, e também impresso em tela as variáveis dos contadores que indicam os anos solicitados (2018,2019,2020 e 2021).

### Capturas de Tela

➔ Código de execução:

```

59 public void opcao2(String nomeMunicipio) {
60
61     String nm = nomeMunicipio;
62     ArrayList<internacoes> lista2 = new ArrayList<>();
63
64     String aux0 = "2017-12-31 23:59:59.000";
65     String aux1 = "2019-01-01 00:00:00.000";
66     String aux2 = "2018-12-31 23:59:59.000";
67     String aux3 = "2020-01-01 00:00:00.000";
68     String aux4 = "2019-12-31 23:59:59.000";
69     String aux5 = "2021-01-01 00:00:00.000";
70     String aux6 = "2020-12-31 23:59:59.000";
71     String aux7 = "2022-01-01 00:00:00.000";
72     try {
73         Date data0 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux0);
74         Date data1 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux1);
75         Date data2 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux2);
76         Date data3 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux3);
77         Date data4 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux4);
78         Date data5 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux5);
79         Date data6 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux6);
80         Date data7 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss.SSS").parse(aux7);
81
82         int cont1=0,cont2=0,cont3=0,cont4=0;
83
84         for(int i=0;i<lista1.size();i++) {
85             if(nm.equalsIgnoreCase(lista1.get(i).getMunicipioResidencia())==true){
86                 if(lista1.get(i).getDataInternacao().after(data0)==true && lista1.get(i).getDataInternacao().before(data7))
87                     lista2.add(lista1.get(i));
88                 if(lista1.get(i).getDataInternacao().after(data0)==true && lista1.get(i).getDataInternacao().before(data1))
89                     cont1++;
90                 if(lista1.get(i).getDataInternacao().after(data2)==true && lista1.get(i).getDataInternacao().before(data3))
91                     cont2++;
92                 if(lista1.get(i).getDataInternacao().after(data4)==true && lista1.get(i).getDataInternacao().before(data5))
93                     cont3++;
94                 if(lista1.get(i).getDataInternacao().after(data6)==true && lista1.get(i).getDataInternacao().before(data7))
95                     cont4++;
96             }
97         }
98         System.out.println("\nLista de pacientes com data de internação entre 2018 e 2021:\n");
99         for(internacoes l : lista2) {
100             System.out.println(l);
101         }
102         System.out.println("\nQuantidade de pacientes internados (por ano):\n\n 2018 -> "+cont1+"\n 2019 -> "+cont2+"\n 2020 -> "
103             +cont3+"\n 2021 -> "+cont4);
104     } catch (ParseException e){
105         System.out.println("ParseException occurred: "+e.getMessage());
106     }
107     System.out.print("\n* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu "
108         + "\n e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).\n");
109 }

```

➔ Resultados

```

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 2
|
-----

Favor informar o nome do município residencial:

```

```

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 2

-----

Favor informar o nome do município residencial: candido godoi

Lista de pacientes com data de internação entre 2018 e 2021:

internacoes [idade=47.0, horasNaFila=144.0, dataExtracao=Mon Aug 30 16:12:44 BRT 2021, dataSolicitacao=Tue Mar 27 00:00:00 BRT 2018, dataAutorizacao=Tue Mar 27 00:00:00 BRT 2018, dataInternacao=Mon Apr 02 00:00:00 BRT 2018, dataAlta=Mon Apr 02 00:00:00 BRT 2018, idUsuario=8cc43df0d0f712427aa6f24598d84e61, situacao=COM_ALTA, centralRegOrigem=REGULACAO HOSPITALAR, sexo=MASCULINO, municipioResidencia=CANDIDO GODOI, solicitacao=PRÁ*PRIA, tipoInternacao=PRÁ*PRIA, tipoLeito=ENFADULTO, executante=IRMANDADE DA SANTA CASA DE MISERICORDIA DE PORTO ALEGRE, codCid=H359, carater=ELETIVA, tipoInternacao=PRÁ*PRIA, tipoLeito=ENFADULTO, executante=HOSPITAL SAO LUCAS DA PUCRS]
internacoes [idade=56.0, horasNaFila=0.0, dataExtracao=Mon Aug 30 16:12:45 BRT 2021, dataSolicitacao=Mon Mar 19 00:00:00 BRT 2018, dataAutorizacao=Mon Mar 19 00:00:00 BRT 2018, dataInternacao=Mon Mar 19 00:00:00 BRT 2018, dataAlta=Mon Mar 19 00:00:00 BRT 2018, idUsuario=1b40d1c8f7c296f1a7e2c887f658fd7, situacao=COM_ALTA, centralRegOrigem=REGULACAO HOSPITALAR, sexo=MASCULINO, municipioResidencia=CANDIDO GODOI, solicitacao=PRÁ*PRIA, tipoInternacao=PRÁ*PRIA, tipoLeito=ENFADULTO, executante=HOSPITAL SAO LUCAS DA PUCRS]
internacoes [idade=62.0, horasNaFila=24.0, dataExtracao=Mon Aug 30 16:12:45 BRT 2021, dataSolicitacao=Wed May 16 00:00:00 BRT 2018, dataAutorizacao=Wed May 16 00:00:00 BRT 2018, dataInternacao=Thu May 17 00:00:00 BRT 2018, dataAlta=Thu May 17 00:00:00 BRT 2018, idUsuario=b52e117df70c637bf91aae1aedc1a5, situacao=COM_ALTA, centralRegOrigem=REGULACAO HOSPITALAR, sexo=FEMININO, municipioResidencia=CANDIDO GODOI, solicitacao=PRÁ*PRIA, tipoInternacao=PRÁ*PRIA, tipoLeito=ENFADULTO, executante=HOSPITAL BANCO DE OLHOS DE PORTO ALEGRE]
internacoes [idade=72.0, horasNaFila=2375.0, dataExtracao=Mon Aug 30 16:12:45 BRT 2021, dataSolicitacao=Wed Oct 14 00:00:00 BRT 2020, dataAutorizacao=Wed Oct 14 00:00:00 BRT 2020, dataInternacao=Thu Jan 21 00:00:00 BRT 2021, dataAlta=Thu Jan 21 00:00:00 BRT 2021, idUsuario=1cbd0bb632561cae8f8bf148c82312756, situacao=COM_ALTA, centralRegOrigem=REGULACAO HOSPITALAR, sexo=MASCULINO, municipioResidencia=CANDIDO GODOI, solicitacao=PRÁ*PRIA, tipoInternacao=PRÁ*PRIA, tipoLeito=ENFADULTO, executante=HOSPITAL BANCO DE OLHOS DE PORTO ALEGRE]
internacoes [idade=67.0, horasNaFila=24.0, dataExtracao=Mon Aug 30 16:12:45 BRT 2021, dataSolicitacao=Tue May 29 00:00:00 BRT 2018, dataAutorizacao=Tue May 29 00:00:00 BRT 2018, dataInternacao=Wed May 30 00:00:00 BRT 2018, dataAlta=Wed May 30 00:00:00 BRT 2018, idUsuario=9fb98a655dbd85c73deb67efcacab680, situacao=COM_ALTA, centralRegOrigem=REGULACAO HOSPITALAR, sexo=MASCULINO, municipioResidencia=CANDIDO GODOI, solicitacao=PRÁ*PRIA, tipoInternacao=PRÁ*PRIA, tipoLeito=ENFADULTO, executante=HOSPITAL BANCO DE OLHOS DE PORTO ALEGRE]
internacoes [idade=67.0, horasNaFila=772.0, dataExtracao=Mon Aug 30 16:12:45 BRT 2021, dataSolicitacao=Mon Apr 29 00:00:00 BRT 2019, dataAutorizacao=Mon Apr 29 00:00:00 BRT 2019, dataInternacao=Mon May 27 00:00:00 BRT 2019, dataAlta=Mon May 27 00:00:00 BRT 2019, idUsuario=deca44cbe2f6bb8e6ad8926857d2fa1d, situacao=COM_ALTA, centralRegOrigem=REGULACAO HOSPITALAR, sexo=FEMININO, municipioResidencia=CANDIDO GODOI, solicitacao=PRÁ*PRIA, tipoInternacao=PRÁ*PRIA, tipoLeito=ENFADULTO, executante=IRMANDADE DA SANTA CASA DE MISERICORDIA DE PORTO ALEGRE]
internacoes [idade=70.0, horasNaFila=672.0, dataExtracao=Mon Aug 30 16:12:45 BRT 2021, dataSolicitacao=Mon Jul 08 00:00:00 BRT 2019, dataAutorizacao=Mon Jul 08 00:00:00 BRT 2019, dataInternacao=Mon Aug 05 00:00:00 BRT 2019, dataAlta=Mon Aug 05 00:00:00 BRT 2019, idUsuario=1cbd0bb632561cae8f8bf148c82312756, situacao=COM_ALTA, centralRegOrigem=REGULACAO HOSPITALAR, sexo=MASCULINO, municipioResidencia=CANDIDO GODOI, solicitacao=PRÁ*PRIA, tipoInternacao=PRÁ*PRIA, tipoLeito=ENFADULTO, executante=HOSPITAL NOSSA SENHORA DA CONCEICAO SA]

Quantidade de pacientes internados (por ano):

2018 -> 4
2019 -> 2
2020 -> 0
2021 -> 1

* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 1

```

## Testes

- Aqui também foram utilizados teste manuais, foram impressas informações e tela para garantir o funcionamento das operações (até que o valor esperado fosse mostrado), além da utilização do comando “try-catch” que sinalizava erros de conversão.

- Em todas os resultados, uma mensagem é apresentada instruindo o usuário a tentar novamente evitando caracteres especiais ou acentuação.

```

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 2

-----

Favor informar o nome do município residencial: vdsvsd

Lista de pacientes com data de internação entre 2018 e 2021:

Quantidade de pacientes internados (por ano):

2018 -> 0
2019 -> 0
2020 -> 0
2021 -> 0

* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada:

```

### FUNCIONALIDADE 3

#### Implementação

- No método `opcao3()`, o usuário fornece uma “instituição de saúde” como parâmetro do tipo `String`;
- Um novo `ArrayList` tipo `String` é criado para armazenar as informações que serão impressas em tela;
- Utilizamos um “for” para percorrer a lista, já no comando “if” todos os objetos/linhas que obedecerem à informação fornecida pelo usuário terão os seus “ids do pacientes, idade, município residencial e solicitante, as datas de autorização, internação e alta, e o executante” armazenadas no novo `ArrayList` “lista3”;
- As variáveis do tipo “Date” necessitaram ser convertidas para o tipo `String` durante o armazenamento (utilizado o método `toString()`);
- O `ArrayList` de `Strings` “lista3” é impresso em tela utilizando um comando “for”.

#### Capturas de Tela

➔ Código de execução:

```

113 public void opcao3(String nomeExecutante) {
114     String nm = nomeExecutante;
115     ArrayList<String> lista3 = new ArrayList<>();
116
117     for(int i=0;i<lista1.size();i++) {
118         if(nm.equalsIgnoreCase(lista1.get(i).getExecutante())==true){
119             lista3.add("ID PACIENTE -> "+lista1.get(i).getIdUsuario()+" ; "+
120                 "IDADE = "+lista1.get(i).getIdade().toString()+" ; "+
121                 "MUNICÍPIO RESIDENCIAL -> "+lista1.get(i).getMunicipioResidencia()+" ; "+
122                 "MUNICÍPIO SOLICITANTE -> "+lista1.get(i).getMunicipioSolicitante()+" ; "+
123                 "DATA DA AUTORIZAÇÃO -> "+lista1.get(i).getDataAutorizacao().toString()+" ; "+
124                 "DATA DA INTERNAÇÃO -> "+lista1.get(i).getDataInternacao().toString()+" ; "+
125                 "DATA DA ALTA -> "+lista1.get(i).getDataAlta().toString()+" ; "+
126                 "EXECUTANTE -> "+lista1.get(i).getExecutante());
127         }
128     }
129
130     System.out.println("\nLista de todos os pacientes que foram internados, sua idade, o município residencial e solicitante de "
131         + " cada um deles, as datas de autorização, de internação e alta e o executante:\n");
132     for(String m : lista3) {
133         System.out.println(m);
134     }
135     System.out.print("\n\n* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu "
136         + "\n e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).\n");
137 }

```

➔ Resultados

```

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 3
|
-----

Favor informar o nome da instituição de saúde executante:

```

```
----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 3

-----

Favor informar o nome da instituição de saúde executante: santa casa de uruguaiana

Lista de todos os pacientes que foram internados, sua idade, o município residencial e solicitante de cada um deles, as datas de autorização, de internação e o executante:

ID PACIENTE -> b5696560b2cfd068e169c6ca3a54a35 ; IDADE = 50.0 ; MUNICÍPIO RESIDENCIAL -> ALEGRETE ; MUNICÍPIO SOLICITANTE -> ALEGRETE ; DATA DA AUTORIZAÇÃO -> Thu Jan 07 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Thu Jan 07 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> 85586a352bc0968cb6fe5fe44008768b ; IDADE = 53.0 ; MUNICÍPIO RESIDENCIAL -> URUGUAIANA ; MUNICÍPIO SOLICITANTE -> URUGUAIANA ; DATA DA AUTORIZAÇÃO -> Tue May 04 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Wed May 05 00:00:00 BRT 2021 ; DATA DA ALTA -> Fri May 28 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> 397bcf39aa3dda4a120662a75d19ec33 ; IDADE = 82.0 ; MUNICÍPIO RESIDENCIAL -> URUGUAIANA ; MUNICÍPIO SOLICITANTE -> URUGUAIANA ; DATA DA AUTORIZAÇÃO -> Fri May 07 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Mon May 10 00:00:00 BRT 2021 ; DATA DA ALTA -> Tue Jun 01 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> d1405ac3b17560b75f7b9682dd67e418 ; IDADE = 74.0 ; MUNICÍPIO RESIDENCIAL -> URUGUAIANA ; MUNICÍPIO SOLICITANTE -> URUGUAIANA ; DATA DA AUTORIZAÇÃO -> Fri May 07 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Mon May 10 00:00:00 BRT 2021 ; DATA DA ALTA -> Tue Jun 01 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> 55620341612d1317463cc2901d6f4c45 ; IDADE = 61.0 ; MUNICÍPIO RESIDENCIAL -> URUGUAIANA ; MUNICÍPIO SOLICITANTE -> URUGUAIANA ; DATA DA AUTORIZAÇÃO -> Tue Jan 12 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Tue Jan 12 00:00:00 BRT 2021 ; DATA DA ALTA -> Sun Jan 24 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> 8ca389f2032acd9d8e979dfe6110152 ; IDADE = 65.0 ; MUNICÍPIO RESIDENCIAL -> ROSARIO DO SUL ; MUNICÍPIO SOLICITANTE -> ROSARIO DO SUL ; DATA DA AUTORIZAÇÃO -> Sun Sep 13 00:00:00 BRT 2020 ; DATA DA INTERNAÇÃO -> Tue Sep 15 00:00:00 BRT 2020 ; DATA DA ALTA -> Wed Sep 16 00:00:00 BRT 2020 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> 1965bf2b7d6e8ba49373e336f8dfe06e ; IDADE = 19.0 ; MUNICÍPIO RESIDENCIAL -> SANTANA DO LIVRAMENTO ; MUNICÍPIO SOLICITANTE -> SANTANA DO LIVRAMENTO ; DATA DA AUTORIZAÇÃO -> Fri Jan 29 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Sat Jan 30 00:00:00 BRT 2021 ; DATA DA ALTA -> Mon Jul 12 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> 8d2eaf183fb160656310ac84daf102a3 ; IDADE = 76.0 ; MUNICÍPIO RESIDENCIAL -> ROSARIO DO SUL ; MUNICÍPIO SOLICITANTE -> ROSARIO DO SUL ; DATA DA AUTORIZAÇÃO -> Thu Sep 24 00:00:00 BRT 2020 ; DATA DA INTERNAÇÃO -> Fri Sep 25 00:00:00 BRT 2020 ; DATA DA ALTA -> Sat Oct 24 00:00:00 BRT 2020 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> e11e9f64f6db6df8ede70b19f2251668 ; IDADE = 81.0 ; MUNICÍPIO RESIDENCIAL -> ROSARIO DO SUL ; MUNICÍPIO SOLICITANTE -> ROSARIO DO SUL ; DATA DA AUTORIZAÇÃO -> Mon Jan 04 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Mon Jan 04 00:00:00 BRT 2021 ; DATA DA ALTA -> Wed Jan 27 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> 70d91a3bd46ab49a11770f0ccc6a50fb ; IDADE = 27.0 ; MUNICÍPIO RESIDENCIAL -> SANTANA DO LIVRAMENTO ; MUNICÍPIO SOLICITANTE -> SANTANA DO LIVRAMENTO ; DATA DA AUTORIZAÇÃO -> Mon Jan 04 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Mon Jan 04 00:00:00 BRT 2021 ; DATA DA ALTA -> Wed Jan 27 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ID PACIENTE -> f8d4d490fb2e4b7607dcfc89e532a9c7 ; IDADE = 78.0 ; MUNICÍPIO RESIDENCIAL -> URUGUAIANA ; MUNICÍPIO SOLICITANTE -> URUGUAIANA ; DATA DA AUTORIZAÇÃO -> Mon Jan 04 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Mon Jan 04 00:00:00 BRT 2021 ; DATA DA ALTA -> Wed Jan 27 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA

* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: |
```

```
ernação e alta e o executante:

ZACÃO -> Thu Jan 07 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Thu Jan 07 00:00:00 BRT 2021 ; DATA DA ALTA -> Thu Jan 21 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
TORIZAÇÃO -> Wed May 05 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Thu May 06 00:00:00 BRT 2021 ; DATA DA ALTA -> Thu May 27 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
TORIZAÇÃO -> Tue May 04 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Wed May 05 00:00:00 BRT 2021 ; DATA DA ALTA -> Fri May 28 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
TORIZAÇÃO -> Fri May 07 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Mon May 10 00:00:00 BRT 2021 ; DATA DA ALTA -> Tue Jun 01 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
TORIZAÇÃO -> Fri May 07 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Mon May 10 00:00:00 BRT 2021 ; DATA DA ALTA -> Tue Jun 01 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ATA DA AUTORIZAÇÃO -> Tue Jan 12 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Tue Jan 12 00:00:00 BRT 2021 ; DATA DA ALTA -> Sun Jan 24 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
IVRAMENTO ; DATA DA AUTORIZAÇÃO -> Wed Jan 20 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Thu Jan 21 00:00:00 BRT 2021 ; DATA DA ALTA -> Thu Feb 04 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ATA DA AUTORIZAÇÃO -> Sun Sep 13 00:00:00 BRT 2020 ; DATA DA INTERNAÇÃO -> Tue Sep 15 00:00:00 BRT 2020 ; DATA DA ALTA -> Wed Sep 16 00:00:00 BRT 2020 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
ATA DA AUTORIZAÇÃO -> Fri Jan 29 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Sat Jan 30 00:00:00 BRT 2021 ; DATA DA ALTA -> Mon Jul 12 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
IVRAMENTO ; DATA DA AUTORIZAÇÃO -> Thu Sep 24 00:00:00 BRT 2020 ; DATA DA INTERNAÇÃO -> Fri Sep 25 00:00:00 BRT 2020 ; DATA DA ALTA -> Sat Oct 24 00:00:00 BRT 2020 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
TORIZAÇÃO -> Mon Jan 04 00:00:00 BRT 2021 ; DATA DA INTERNAÇÃO -> Mon Jan 04 00:00:00 BRT 2021 ; DATA DA ALTA -> Wed Jan 27 00:00:00 BRT 2021 ; EXECUTANTE -> SANTA CASA DE URUGUAIANA
```

## Testes

- Foram utilizados testes manuais da mesma forma que as funcionalidades anteriores;

- Em todas os resultados, uma mensagem é apresentada instruindo o usuário a tentar novamente evitando caracteres especiais ou acentuação.

```
----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 3

-----

Favor informar o nome da instituição de saúde executante: hsght

Lista de todos os pacientes que foram internados, sua idade, o município residencial e solicitante de

* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu
e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada:
```

## FUNCIONALIDADE 4

### Implementação

- No método opcao4() recebemos o nome da “instituição de saúde” fornecida pelo usuário;
- É criado um novo objeto do tipo ArrayList de Strings chamado de “lista4”;
- Em um comando “for” percorremos a lista principal, utilizamos um “if” para identificar se a instituição de saúde fornecida pelo usuário foi correspondência na lista, caso exista valor correspondente, serão adicionados a nova “lista4” os campos “id do paciente, hospital executante e o tempo de internação em dias”;
- Para calcular o tempo de internação em dias foi necessário armazenar em uma variável “aux” do tipo “long” a diferença entre a “data de alta” do paciente e a “data de solicitação”, para este calculo foi necessário transformar ambas as datas para a mesma unidade utilizando o método getTime() que converte a data para “milissegundos”;
- Calculada a diferença entre as datas em milissegundos, o valor é dividido por “86400000L” (para termos o resultado em dias) e é armazenado na variável “diasDeInternação”;
- O ArrayList de Strings “lista4” é impresso em tela utilizando um comando “for”.

### Capturas de Tela

➔ Código de execução:



```

138• public void opcao4(String nomeSolicitante) {
139
140     String nm = nomeSolicitante;
141     ArrayList<String> lista4 = new ArrayList<>();
142     long aux = 0;
143     long diasDeInternacao = 0;
144
145     for(int i=0;i<lista1.size();i++) {
146         if(nm.equalsIgnoreCase(lista1.get(i).getSolicitante())==true){
147
148             aux = (lista1.get(i).getDataAlta().getTime() - lista1.get(i).getDataSolicitacao().getTime());
149             diasDeInternacao = (aux / 86400000L);
150             lista4.add("ID DO PACIENTE -> "+lista1.get(i).getIdUsuario()+"; HOSPITAL EXECUTANTE -> "+lista1.get(i).getExecutante()+
151                 "; TEMPO DE INTERNAÇÃO -> "+String.valueOf(diasDeInternacao)+" DIAS");
152         }
153     }
154
155     System.out.println("\nLista de todos os pacientes que foram internados, sua idade, o município residencial e solicitante de"
156         + "\n cada um deles, as datas de autorização, de internação e alta e o executante:\n");
157     for(String n : lista4) {
158         System.out.println(n);
159     }
160     System.out.print("\n* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu "
161         + "\n e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).\n");
162 }
163

```

## ➔ Resultados

```

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 4

-----

Favor informar o nome da instituição de saúde solicitante: santa casa de uruguaiana

Lista de todos os pacientes que foram internados, sua idade, o município residencial e solicitante de
cada um deles, as datas de autorização, de internação e alta e o executante:

ID DO PACIENTE -> ff07e6ad8e5039faa281c094d6f08514; HOSPITAL EXECUTANTE -> HOSPITAL AUXILIADORA; TEMPO DE INTERNAÇÃO -> 225 DIAS
ID DO PACIENTE -> 85586a352bc0968cb6fe5fe44008768b; HOSPITAL EXECUTANTE -> SANTA CASA DE URUGUAIANA; TEMPO DE INTERNAÇÃO -> 22 DIAS
ID DO PACIENTE -> 397bcf39aa3dda4a120662a75d19ec33; HOSPITAL EXECUTANTE -> SANTA CASA DE URUGUAIANA; TEMPO DE INTERNAÇÃO -> 24 DIAS
ID DO PACIENTE -> d1405ac3b17560b75f7b9682dd67e418; HOSPITAL EXECUTANTE -> SANTA CASA DE URUGUAIANA; TEMPO DE INTERNAÇÃO -> 25 DIAS
ID DO PACIENTE -> 55620341612d1317463cc2901d6f4c45; HOSPITAL EXECUTANTE -> SANTA CASA DE URUGUAIANA; TEMPO DE INTERNAÇÃO -> 25 DIAS
ID DO PACIENTE -> 52f64f283481ffe0d460d7ffe384a728; HOSPITAL EXECUTANTE -> HOSPITAL SAO LUIZ; TEMPO DE INTERNAÇÃO -> 9 DIAS
ID DO PACIENTE -> ae828ee811022ba096805590efd36057; HOSPITAL EXECUTANTE -> SANTA CASA DE SAO GABRIEL; TEMPO DE INTERNAÇÃO -> 6 DIAS
ID DO PACIENTE -> 08e245c2f41938f101f0f586c3e82f5f; HOSPITAL EXECUTANTE -> HOSPITAL SAO LUIZ; TEMPO DE INTERNAÇÃO -> 2 DIAS
ID DO PACIENTE -> bdeb5c8c18d8d333c49b341369a34c51; HOSPITAL EXECUTANTE -> HOSPITAL SAO LUIZ; TEMPO DE INTERNAÇÃO -> 2 DIAS
ID DO PACIENTE -> f8d4d490fb2e4b7607dcfc89e532a9c7; HOSPITAL EXECUTANTE -> SANTA CASA DE URUGUAIANA; TEMPO DE INTERNAÇÃO -> 23 DIAS

* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu
e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada:

```

## Testes

- Foram utilizados testes manuais da mesma forma que as funcionalidades anteriores;
- Em todas os resultados, uma mensagem é apresentada instruindo o usuário a tentar novamente evitando caracteres especiais ou acentuação.

```
----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa

Digite a opção desejada: 4

-----

Favor informar o nome da instituição de saúde solicitante: 165h1g6j41gh6j56g

Lista de todos os pacientes que foram internados, sua idade, o município residencial e solicitante de
cada um deles, as datas de autorização, de internação e alta e o executante:

* Caso você não tenha encontrado a informação desejada, por favor selecione novamente a opção no menu
e forneça novos valores de entrada (evite o uso de caracteres especiais ou acentuação).

----- SEJA BEM-VINDO(A) -----

1. Consultar média de idade dos pacientes
2. Consultar internações por ano
3. Consultar hospitais
4. Calcular tempo de internação
5. Determinar tempos de espera na fila
6. Terminar o programa
```

## FUNCIONALIDADE 5

### Implementação

- No método `opcao5()` temos a execução de comandos “for”, estes comandos efetuam comparações sucessivas para, utilizando o comando “if”, determinar o maior elemento da lista principal (compara um a um dos elementos do Array), quando o maior elemento é encontrado, este é armazenado na variável “maiorValor”;

- No comando “for” seguinte, a variável “maiorValor” é localizada e sem seguida, através do método `setHorasNaFila()`, ela é zerada para que possamos localizar o segundo maior valor no comando “for” seguinte;

- Na impressão dos dados em tela foi utilizado o “DecimalFormat” para tirarmos o “.0” das variáveis “Double”.

### Capturas de Tela

➔ Código de execução:



```
164 public void opcao5() {
165
166     Double maiorValor=0.0;
167     Double aux0,aux1,aux2,aux3,aux4;
168
169     for(int i=0;i<lista1.size();i++) {
170         if(lista1.get(i).getHorasNaFila() > maiorValor)
171             maiorValor = lista1.get(i).getHorasNaFila();
172     }
173     aux0 = maiorValor;
174     for(int i=0;i<lista1.size();i++) {
175         if(lista1.get(i).getHorasNaFila() == maiorValor)
176             lista1.get(i).setHorasNaFila(0.0);
177     }
178     maiorValor = 0.0;
179     for(int i=0;i<lista1.size();i++) {
180         if(lista1.get(i).getHorasNaFila() > maiorValor)
181             maiorValor = lista1.get(i).getHorasNaFila();
182     }
183     aux1 = maiorValor;
184     for(int i=0;i<lista1.size();i++) {
185         if(lista1.get(i).getHorasNaFila() == maiorValor)
186             lista1.get(i).setHorasNaFila(0.0);
187     }
188     maiorValor = 0.0;
189     for(int i=0;i<lista1.size();i++) {
190         if(lista1.get(i).getHorasNaFila() > maiorValor)
191             maiorValor = lista1.get(i).getHorasNaFila();
192     }
193     aux2 = maiorValor;
194     for(int i=0;i<lista1.size();i++) {
195         if(lista1.get(i).getHorasNaFila() == maiorValor)
196             lista1.get(i).setHorasNaFila(0.0);
197     }
198     maiorValor = 0.0;
199     for(int i=0;i<lista1.size();i++) {
200         if(lista1.get(i).getHorasNaFila() > maiorValor)
201             maiorValor = lista1.get(i).getHorasNaFila();
202     }
203     aux3 = maiorValor;
204     for(int i=0;i<lista1.size();i++) {
205         if(lista1.get(i).getHorasNaFila() == maiorValor)
206             lista1.get(i).setHorasNaFila(0.0);
207     }
208     maiorValor = 0.0;
209     for(int i=0;i<lista1.size();i++) {
210         if(lista1.get(i).getHorasNaFila() > maiorValor)
211             maiorValor = lista1.get(i).getHorasNaFila();
212     }
213     aux4 = maiorValor;
214     for(int i=0;i<lista1.size();i++) {
215         if(lista1.get(i).getHorasNaFila() == maiorValor)
216             lista1.get(i).setHorasNaFila(0.0);
217     }
218     DecimalFormat format = new DecimalFormat("##");
219     System.out.println("\nOs 5 maiores tempos de espera da lista (em horas) são:\n 1º -> "+format.format(aux0)+
220     "\n 2º -> "+format.format(aux1)+"\n 3º -> "+format.format(aux2)+"\n 4º -> "+format.format(aux3)+"\n 5º -> "+format.format(aux4));
221 }
```

➔ Resultados

```
----- SEJA BEM-VINDO(A) -----  
  
    1. Consultar média de idade dos pacientes  
    2. Consultar internações por ano  
    3. Consultar hospitais  
    4. Calcular tempo de internação  
    5. Determinar tempos de espera na fila  
    6. Terminar o programa  
  
Digite a opção desejada: 5  
|  
-----  
  
Os 5 maiores tempos de espera da lista (em horas) são:  
  
1º -> 876599  
2º -> 43824  
3º -> 34176  
4º -> 31463  
5º -> 29543  
  
----- SEJA BEM-VINDO(A) -----  
  
    1. Consultar média de idade dos pacientes  
    2. Consultar internações por ano  
    3. Consultar hospitais  
    4. Calcular tempo de internação  
    5. Determinar tempos de espera na fila  
    6. Terminar o programa  
  
Digite a opção desejada:
```

### Testes

- Já que este método não recebe parâmetros, os testes manuais se restringiram a algumas poucas impressões do programa ("System.out.print") durante a escrita do código.

## **FUNCIONALIDADE 6**

### Implementação

- O método opcoes6() apenas direciona ao comando "System.exit(0)", este que é responsável por encerrar o programa;

- Em tela a mensagem de encerramento do programa é mostrada e o menu não retorna a aparecer.

### Capturas de Tela

➔ Código de execução:

```
224  
225 • public void opcao6() {  
226  
227     System.out.println("Programa finalizado com sucesso !");  
228     System.exit(0);  
229 }  
230
```

➔ Resultados

```
----- SEJA BEM-VINDO(A) -----  
  
1. Consultar média de idade dos pacientes  
2. Consultar internações por ano  
3. Consultar hospitais  
4. Calcular tempo de internação  
5. Determinar tempos de espera na fila  
6. Terminar o programa  
  
Digite a opção desejada: 6  
  
-----  
Programa finalizado com sucesso !
```

Testes

- Neste método não houve a necessidade de teste.

# OBRIGADO !

...e me chama.

