



**UNIVERSIDADE CATÓLICA DE ANGOLA**  
**FACULDADE DE ENGENHARIA**

---

**Curso de Engenharia Informática**  
**Curso de Engenharia de Telecomunicações**

**Ano Lectivo 2016**

**3º Ano – 2º Semestre**

**Sistemas Distribuídos e Paralelos(I)**

**Exame de Recurso A**

**Data:** 15-Janeiro-2016

**Professor:** Prof. Dr. Aires Veloso

**Duração:** 105 minutos

**Tolerância:** 15 minutos

**Nota:** Siga as seguintes sugestões, nomeadamente:

- apresente sempre letra legível(letra de imprensa, de preferência).
- apresente sempre o desenho("draft") do interface gráfico(i.e. todas as janelas) do programa.
- enfatize a arquitectura da aplicação. Siga uma metodologia descendente("top-down"), i.e. identifique e implemente as principais entidades; relativamente as entidades de suporte, em alguns casos dispense a implementação dos métodos e apresente apenas o protótipo e a descrição com a semântica do método.
- apresente o script da base de dados.

A energia eléctrica é um recurso chave para o desenvolvimento. Nesta conformidade, o governo angolano reestruturou o sector eléctrico e criou 3 empresas:

- uma para gerir os subsistemas de produção de energia (1º nível);
- outra para gerir os subsistemas de transporte de energia (2º nível);
- uma terceira para gerir os subsistemas de distribuição de energia (3º nível);

Considere os seguintes requisitos:

- os diferentes subsistemas funcionam em cadeia numa lógica fornecedor-cliente, i.e. os

subsistemas de transporte de energia transportam a energia disponibilizada pelos subsistemas de produção e os subsistemas de distribuição de energia transportam para os consumidores finais (empresas, residências, etc), a energia disponibilizada pelos subsistemas de transporte;

- que existem sistemas de produção de energia hídrica, térmica, eólica e solar;
- que apenas algumas províncias detêm sistemas de produção de energia e que só essas podem deter sistemas de transporte de energia;
- todas as províncias consomem energia e detêm subsistemas de distribuição de energia;

Considere que está a desenvolver um sistema de gestão (Web) para o Ministério de Energia e Águas. Numa primeira fase, considere que está apenas interessado em controlar a eficiência do sistema e dos seus componentes e por conseguinte pretende monitorar os seguintes parâmetros ao longo de cada mês:

- o potencial de energia máximo (de produção, de transporte e de distribuição) de cada subsistema em megawatts (PEM);
- o potencial de energia efectiva em megawatts (PEE) que cada subsistema disponibilizou aos seus clientes que é condicionado por vários factores, nomeadamente: o consumo efectivo dos consumidores finais, as avarias e o tempo médio de recuperação de cada avaria; é de referenciar que em cada subsistema, a PEE não pode superar a PEM;
- a eficiência(EF) é fornecida pela seguinte relação:  $EF = PEE \times 100 / PEM$

Faça um script que lança a base de dados(SQL).

```
DROP DATABASE IF EXISTS sectorelectricodb ;
CREATE DATABASE sectorelectricodb ;
USES sectorelectricodb ;

--
-- Table structure for table provincias
--

CREATE TABLE provincias (
    pk_provincia SERIAL PRIMARY KEY,
    designacao varchar(45) NOT NULL,
);

INSERT INTO provincias (designacao)
VALUES ('Cabinda'), ('Zaire'), ('Luanda'), ('Uige'), ('Lunda-
Norte'), ('Lunda-Sul'), ('Kwanza-Norte');

CREATE TABLE tipo_energia (
    pk_tipo_energia SERIAL PRIMARY KEY,
    designacao varchar(45) NOT NULL,
);

INSERT INTO tipo_energia (designacao)
VALUES ('hidrica'), ('termica'), ('eolica'), ('solar');

--
-- Table structure for table sistemas_producao
--

CREATE TABLE sistemas_producao (
    pk_sistema_producao SERIAL PRIMARY KEY,
    designacao varchar(45) NOT NULL,
```

```

        fk_tipo_energia      int(11) NOT NULL,
        fk_provincia         int(11) NOT NULL,
        pem                  int(11) NOT NULL,

        FOREIGN KEY (fk_tipo_energia) REFERENCES
                                tipo_energia (pk_tipo_energia)
        FOREIGN KEY (fk_provincia) REFERENCES
                                provincias (pk_provincia)
);

--
-- Table structure for table  sistemas_transporte
--

CREATE TABLE  sistemas_transporte (
        pk_sistema_transporte  SERIAL PRIMARY KEY,

        designacao              varchar(45) NOT NULL,
        pem                    int(11) NOT NULL,
);

--
-- Table structure for table  sistemas_de_producao_transporte
-- um sub-sistema de transporte pode estar associado a varios
sub-
-- sistemas de producao
--

CREATE TABLE  sistemas_de_producao_transporte (
        pk_sistemas_de_producao_transporte  int(11) NOT NULL
                                                AUTO_INCREMENT,
        fk_sistema_producao                  int(11) NOT NULL
        fk_sistema_transporte                int(11) NOT NULL,

        FOREIGN KEY (fk_sistema_producao) REFERENCES
                                sistemas_producao (pk_sistema_producao)
        FOREIGN KEY (fk_sistema_transporte) REFERENCES
                                sistemas_transporte (pk_sistema_transporte)
);

--
-- Table structure for table  sistemas_distribuicao
--

CREATE TABLE  sistemas_distribuicao (
        pk_sistema_distribuicao  SERIAL PRIMARY KEY
        designacao              varchar(45) NOT NULL,
        pem                    int(11) NOT NULL,
        fk_provincia            int(11) NOT NULL,
        FOREIGN KEY (fk_provincia) REFERENCES
                                provincias (pk_provincia)
);

--
-- Table structure for table  sistemas_de_transporte_distribuicao
-- um sub-sistema de distribuicao pode estar associado a varios
sub-sistemas de transporte
--

```

```

CREATE TABLE sistemas_de_transporte_distribuicao (
    PK_sistemas_de_transporte_distribuicao SERIAL PRIMARY KEY,
    fk_sistema_distribuicao int(11) NOT NULL
fk_sistema_transporte int(11) NOT NULL,

    FOREIGN KEY (fk_sistema_distribuicao) REFERENCES
    sistemas_distribuicao (pk_sistema_distribuicao)

    FOREIGN KEY (fk_sistema_transporte) REFERENCES
    sistemas_transporte (pk_sistema_transporte)
);

--
-- Table structure for table tipo_sistemas
--

CREATE TABLE tipo_sistemas (
    pk_tipo_sistema SERIAL PRIMARY KEY,
    designacao varchar(45) NOT NULL,
);

INSERT INTO tipo_sistemas (designacao)
VALUES ('producao'), ('transporte'), ('distribuicao');

--
-- Table structure for table potencia_energetica_efectiva
-- regista a energia efectiva consumida ao longo dos meses
-- para os todos sub-sistemas
--

CREATE TABLE potencia_energetica_efectiva (
    pk_potencia_energetica_efectiva SERIAL PRIMARY KEY,
    fk_tipo_sistema int(11) NOT NULL
    fk_sistema int(11) NOT NULL,

    mes int(2) NOT NULL,
    ano int(11) NOT NULL,
    pee int(11) NOT NULL,

    FOREIGN KEY (fk_tipo_sistema) REFERENCES
    tipo_sistemas (pk_tipo_sistema),

    FOREIGN KEY (fk_sistema) REFERENCES sistemas
    (pk_sistema)

);

```

**(tempo de resolução: 25 minutos)**  
**(cotação: 5 valores)**

- a) Faça uma primeira versão do código da aplicação que possibilita ao utilizador saber qual a província que em determinado intervalo de tempo, teve maior ineficiência no sistema de distribuição e qual a que teve maior eficiência nos sistemas de produção.

```

<!-- consulta.jsp -->
<form action="processarConsulta">

```

```

<p>
    Introduza as 2 datas(ano/mes)
</p>
<table>
    <tr>
        <td> Ano 1: </td>
        <td>
            <input type = "text" name = "ano1" />
        </td>

        <td> Mes 1: </td>
        <td>
            <select name="mes1" size="1">
                <option value="1">Janeiro</ option>
                <option value="2">Fevereiro</option>
                <option value="3">Marco</option>
                <option value="4">Abril</option>
                <!-- ... -->
                <option value="12">Dezembro</option>
            </select>
        </td>
    </tr>

    <tr>
        <td> Ano 2: </td>
        <td>
            <input type = "text" name = "ano2" />
        </td>

        <td> Mes 2: </td>
        <td>
            <select name="mes2" size="1">
                <option value="1">Janeiro</ option>
                <option value="2">Fevereiro</option>
                <option value="3">Marco</option>
                <option value="4">Abril</option>
                <!-- ... -->
                <option value="12">Dezembro</option>
            </select>
        </td>
    </tr>

    <tr>
        <td colspan = "4">
            <input type = "submit" value = "Submit" />
        </td>
    </tr>
</table>
</form>

```

A entidade "processarConsulta" corresponde a servlet "ProcessarConsulta.java".

// ficheiro ProcessarConsulta.java

```

public class ProcessarConsulta extends HttpServlet
{

```

```

protected void doGet(HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException, IOException
{
    Data data1 = new Data(
        request.getParameter("ano1"),
        request.getParameter("mes1"));

    Data data2 = new Data(
        request.getParameter("ano2"),
        request.getParameter("mes2"));
    processar(data1, data2);
}

public void processar(Data data1, Data data2)
{
    String provMaisInefDistribuicao =
        getProvMaisInefDistribuicao(
            data1, data2);
    String provMaisEficProducao =
        getProvMaisEficProducao(
            data1, data2);
    response.sendRedirect("relatorio?" +
        "provEficiente=" + provMaisEficProducao +
        ",provIneficiente=" +
            provMaisInefDistribuicao);
}

public String getProvMaisInefDistribuicao(Data data1, Data
data2)
{
    /* retorna a provincia com o menor valor medio
        de eficiencia(PEE * 100 / PEM) nos sistemas de
        distribuicao */
}

public String getProvMaisEficProducao(Data data1, Data
data2)
{
    /* retorna a provincia com o maior valor medio
        de eficiencia(PEE * 100 / PEM) nos sistemas de
        producao */
}
} // fim da servlet ProcessarConsulta

```

O objecto "relatorio" é a servlet Relatorio

// ficheiro Relatorio.java

```

public class Relatorio extends HttpServlet
{
    protected void doGet(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException
    {
        String provEficiente =
            request.getParameter("provEficiente");
    }
}

```

```

        String provIneficiente =
            request.getParameter("provIneficiente");

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html><head>");
        out.println("<title>Resposta a Consulta</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>A provincia com o sistema de
distribuicao mais ineficiente eh: " + provIneficiente +
                                ",<br/>");
        out.println("<h1>A provincia com o sistema de producao
mais eficiente eh: " + provEficiente + "</h1>");
        out.println("</body>");
        out.println("</html>");
        out.close();
    }
} // fim da class Relatorio.java

public class Data
{
    private int ano, mes;
    public Data(String ano, String mes)
    {
        this.ano = Integer.parseInt(ano);
        this.mes = Integer.parseInt(mes);
    }
}

```

**(tempo de resolução: 80 minutos)**  
**(cotação: 15 valores)**