

# 1ª Prova (P1)

20/Setembro/2019

**Curso:** Bacharelado em Ciência da Computação – **BCC- Noturno**  
**Disciplina:** Linguagens de Programação Não Convencionais  
**Professora:** Simone das Graças Domingues Prado  
**Horário:** 19h30-21h30

Questão 01		2,5
Questão 02		0,5
Questão 03		1,0
Questão 04		0,5
Questão 05		3,0
Questão 06		2,5
<b>Total</b>		<b>10,0</b>

**Nome:** \_\_\_\_\_ **RA:** \_\_\_\_\_

**(Questão 01)** Foram modelados os predicados:

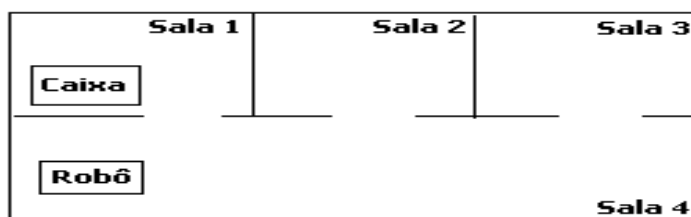
% empregado (NumEmp, NomeEmp, NumDepto). empregado(32,jSilva,21). empregado(74,mReis,25). empregado(89,cMelo,28). empregado(92,rSilva,25). empregado(112,rPinto,21). empregado(121,vSimao,28). empregado(130,jNeves,28).	% salario (NumEmp, Salario). salario(32,380.07). salario(74,400.26). salario(89,520.73). salario(92,480.82). salario(112,390.17). salario(121,905.04). salario(130,640.58).	% departamento (NumDepto, NomeDepto) departamento(21, pessoal). departamento(25, financeiro). departamento(28, tecnico). % ramal(NumDepto, Ramal) ramal(21,142). ramal(25,143). ramal(28,144).
---	--	---

- (a) **(0,5pt)** Crie uma cláusula em Prolog que fornecido o **NomeEmp**, ele devolva o **NomeDepto** e seu **Ramal**.  
**Exemplo: consulta01(jSilva, Departamento, Ramal).**  
**Departamento = pessoal, Ramal = 142**
- (b) **(1,0pt)** Crie uma cláusula em Prolog que fornecido o **NomeDepto**, devolva a lista dos Empregados com seus respectivos salários.  
**Exemplo: consulta02(pessoal, Empregados).**  
**Empregados = [[jSilva, 380.07], [rPinto, 390.17]].**
- (c) **(1,0pt)** Crie uma cláusula em Prolog que devolva o gasto total da empresa com funcionários.  
**Exemplo: consulta03(ValorTotal).**  
**ValorTotal = 3717.67.**

**(Questão 02)** **(0,5pt)** Crie um programa para apagar e devolver o N-ésimo elemento de uma lista  
?- apagar([10, 9, 7, 13], 3, Valor, Lista).  
**Valor = 7, Lista = [10, 9, 13]**

**(Questão 03)** **(1,0pt)** Imagine um cenário onde um robô precise pegar uma caixa que está em uma sala e levá-la até outra sala. Sabe-se que o robô precisa usar as portas entre a sala04 e as salas 1, 2 e 3 para transitar. Desconsidere os passos do robô. Apenas preveja o deslocamento do robô e se ele está com a caixa em mãos. Considerando esse problema e usando os métodos de busca estudados e as informações passadas no enunciado, construa os predicados (estado\_inicial(), meta() e transforma()) para que o robô saia da posição inicial (mostrada na figura acima) e leve a caixa para a sala2 e retorne para a sala1. Se for necessário, construa outros predicados para auxiliar no processo.

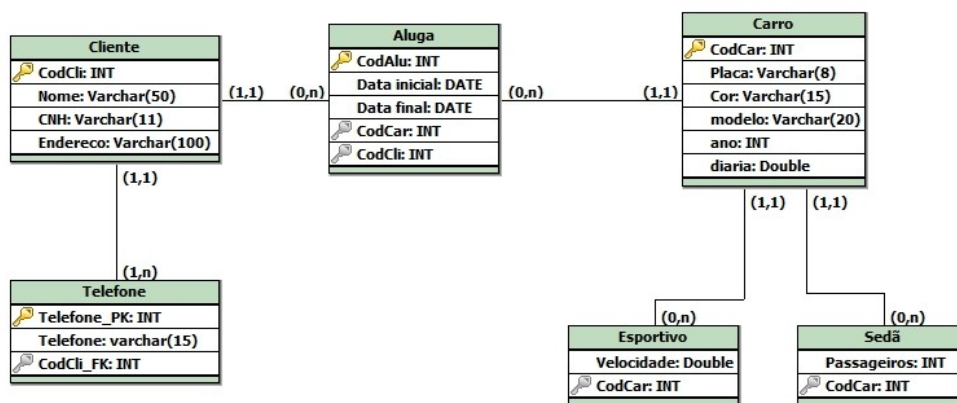
**Sugestão: [Sala\_Em\_Que\_O\_Robo\_Está, Sala\_Em\_Que\_O\_Caixa\_Está, Robo\_Está\_Com\_A\_Caixa?],**  
**então estado\_inicial([sala4,sala1,0]) e meta([sala1,sala2,0]).**



(Questão 04) (0,5pt) Identifique o método de ordenação abaixo:

ordena ([],[ ]).  
ordena ([Cab|Cauda],Lista\_ord):-  
    ordena (Cauda,Cauda\_ord),  
    ordena2 (Cab,Cauda\_ord,Lista\_ord).  
ordena2 (X,[Y|Cauda\_ord1],[Y|Cauda\_ord2]):-  
    X>Y, !, ordena2 (X,Cauda\_ord1,Cauda\_ord2).  
ordena2 (X,Cauda\_ord,[X|Cauda\_ord]).

(Questão 05)



Fonte: <https://pt.stackoverflow.com/questions/291507/banco-de-dados-modelo-l%C3%B3gico-e-conceitual>

```
% cliente(CodCli, Nome, CNH, Endereco).
cliente(1, mariaSilva, 1111, 'Rua 3, 56 - SP').
cliente(2, jairSouza, 2222, 'Rua 3A, 356 - SP').
cliente(3, veraCintra, 3333, 'Rua 3B, 526 - SP').
cliente(4, cristianeJabur, 4444, 'Rua 5, 16 - SP').
cliente(5, dalvaMoreira, 5555, 'Rua 5A, 891 - SP').
```

```
% telefone(Telefone_PK, Telefone, CodCli).
telefone(1, 997155656, 1).
telefone(2, 987155447, 2).
telefone(3, 996472525, 3).
telefone(4, 984756572, 4).
telefone(5, 995149898, 5).
```

```
% aluga(CodAlu, DataInicial, DataFinal, CodCar, CodCli).
aluga(1, data(1,1,2019), data(3,1,2019), 1, 1).
aluga(2, data(3,4,2019), data(7,4,2019), 1, 2).
aluga(3, data(5,5,2019), data(15,5,2019), 2, 3).
aluga(4, data(2,7,2019), data(22,7,2019), 1, 4).
aluga(5, data(7,8,2019), data(28,8,2019), 2, 5).
```

```
% carro(CodCar, Placa, Cor, Modelo, Ano, Diaria).
carro(1, bxf9800, prata, logan, 2017, 139.51).
carro(2, cdx7866, preto, duster, 2018, 240.82).
```

```
% esportivo(CodCar, Velocidade).
esportivo(2, 200).
```

```
% sedan(CodCar, Passageiro).
sedan(1, 5).
```

Tendo como base a figura acima, que representa um modelo de banco de dados, e alguns predicados com informações, pede-se as consultas abaixo:

I) (1,5pt) Dada a placa do carro, descobrir quantas vezes ele foi alugado (total) e para quais clientes (Nome, telefone e endereço).

?- consulta01(bxf9800).

O cliente mariaSilva alugou o carro. Telefone: 997155656. Endereco: Rua 3, 56 - SP.  
O cliente jairSouza alugou o carro. Telefone: 987155447. Endereco: Rua 3A, 356 - SP.  
O cliente veraCintra alugou o carro. Telefone: 996472525. Endereco: Rua 3B, 526 - SP.  
O cliente cristianeJabur alugou o carro. Telefone: 984756572. Endereco: Rua 5, 16 - SP.  
O cliente dalvaMoreira alugou o carro. Telefone: 995149898. Endereco: Rua 5A, 891 - SP.

Quantidade total de alugueis do carro de placa bxf9800: 5

- II) (1,5pt) Sabendo todos os alugueis estão no BDD, calcule o valor que cada cliente gastou. Calcule também o valor que carro obteve. Considere que existe somente os 2 carros que estão na BDD. Considere que para calcular o nro de diárias só se faça a diferença entre os dias (não pense em mês ou ano).

Exemplo:           ?- consulta02(L).  
                  O Cliente 1 pagou R\$ 418.53.  
                  O Cliente 2 pagou R\$ 697.55.  
                  O Cliente 3 pagou R\$ 2649.02.  
                  O Cliente 4 pagou R\$ 2929.71.  
                  O Cliente 5 pagou R\$ 5298.04.  
                  Assim, cada carro teve:  
                  Carro 1: R\$4045.79  
                  Carro 2: R\$7947.06

**Observação:** não precisa construir a impressão das listas.

**(Questão 06)** Tendo em vista o BDD do exercício anterior, vamos torná-lo dinâmico. De uma maneira bem simples podemos construir o programa abaixo (como foi feito em outros exercícios em aula):

```
menu:- nl,
        write('Escolha uma opcao: '),nl,
        write('1. Carrega o arquivo'),nl,
        write('2. Adiciona um cliente'),nl,
        write('3. Adiciona um telefone do cliente'),nl,
        write('4. Adiciona um pedido de aluguel'),nl,
        write('5. Adiciona um carro'),nl,
        write('6. Salva o Banco de Dados em um arquivo'),nl,
        write('7. Termina. '),nl,
        write(' '),nl,
        read(Escolha),
        processa(Escolha),
        menu.

processa(1):- carrega.
processa(2):- adiciona_cliente.
processa(3):- adiciona_telefone.
processa(4):- adiciona_aluga.
processa(5):- adiciona_carro.
processa(6):- salva.
processa(7):- write('Tem certeza que deseja terminar?(s/n)'),
               read(Resposta),Resposta=='s',nl,
               write('Terminado!'),abort.

processa(7):- menu.
```

- I) (1,0pt) Crie a cláusula que faz a adição de um **telefone**(IDTelefone,Telefone,CodCli) sabendo que o **IDTelefone** tem ser único, **CodCli** já tem de existir na BDD. Se isso não ocorrer, apenas diga que não foi possível fazer inserção e encerre a adição do pedido.
- II) (1,0pt) Crie a cláusula que carregue o arquivo ('p1q06.txt') se ele existir. Se não existir só avise que ainda não existe.
- III) (0,5pt) Crie a cláusula que salva em arquivo ('p1q06.txt') os dados do BDD.