

Avaliação Prática de Paradigmas de Linguagens de Programação: Dia 02

Introdução ao Paradigma Lógico

O paradigma lógico de programação é uma abordagem que utiliza a lógica formal para resolver problemas, diferente dos paradigmas imperativo ou funcional que podem focar na sequência de instruções ou na aplicação de funções. Em linguagens lógicas, como Prolog e, mais recentemente, ferramentas que utilizam a biblioteca pyDatalog no Python, a programação é baseada em fatos e regras. Nesse paradigma, o programador define as relações entre os dados, e o mecanismo de inferência da linguagem se encarrega de resolver as consultas com base nesses fatos e regras.

Uso da Biblioteca pyDatalog

A biblioteca pyDatalog permite implementar conceitos do paradigma lógico de programação no ambiente Python, utilizando fatos e regras que se assemelham ao funcionamento do Prolog. Com pyDatalog, é possível modelar dados e realizar consultas complexas de maneira declarativa, deixando que o sistema lide com a lógica subjacente.

Para instalar a biblioteca pyDatalog, é necessário utilizar o gerenciador de pacotes do Python (pip). Execute o seguinte comando em seu terminal para realizar a instalação:

pip install pyDatalog

Esse comando adicionará a biblioteca ao ambiente Python, permitindo importar pyDatalog em seu script e começar a desenvolver consultas e predicados lógicos.

Instruções para a Avaliação Prática

1. Importação dos Fatos (0,5 Pontos)

- Inicie o código importando todos os dados fornecidos nos arquivos funcionarios.txt, alocacoes.txt, departamentos.txt e projetos.txt. Cada arquivo contém informações importantes sobre funcionários, alocações, departamentos e projetos da empresa.
- Cada linha dos arquivos deve ser lida e inserida como fatos na linguagem lógica, definindo relacionamentos como a alocação de funcionários em projetos ou o departamento ao qual cada funcionário pertence.

2. Restrições de Acesso aos Dados

- Após a importação dos dados como fatos, nenhuma das consultas ou operações subsequentes poderá acessar diretamente os arquivos. Todos os predicados, consultas e verificações devem basear-se exclusivamente nos fatos já importados.
- A ideia é garantir que o sistema trabalhe apenas com os fatos lidos dos arquivos, assegurando que qualquer consulta ou operação lógica seja realizada a partir dessas informações, sem necessidade de reabrir os arquivos.

3. Uso dos Nomes de Fatos Solicitados

- Todos os nomes de fatos solicitados nas questões devem ser respeitados conforme indicados nos enunciados (por exemplo, Funcionario, Projeto, Departamento, TrabalhaEm, entre outros). Isso garante a consistência e facilita a interpretação e a correção das respostas.

- Fatos adicionais podem ser criados se necessários para auxiliar a resolução das questões, desde que respeitem a lógica e o contexto dos dados apresentados.
-

Exercício 4: Alocação e Capacidade de Projetos por Departamento (0,5 Pontos)

Instrução: Com base nos dados de alocação e departamento, crie um predicado para verificar se cada departamento está com a capacidade total ou abaixo dela em relação ao número de funcionários alocados em seus projetos.

1. Crie o predicado DepartamentoLotado(D), onde D é o nome do departamento, para identificar departamentos com capacidade máxima de funcionários.
 2. Liste os departamentos com funcionários alocados em projetos e indique quais estão lotados.
 3. Selecione e exiba os funcionários alocados em projetos dos departamentos que não estão lotados.
-

Exercício 5: Encontre os Projetos Compartilhados por Departamentos (1,0 Pontos)

Instrução: Alguns projetos podem envolver múltiplos departamentos, com diferentes equipes alocadas. Seu objetivo é identificar esses projetos e listar todos os funcionários que trabalham neles.

1. Crie o predicado ProjetoCompartilhado(P), onde P é o nome do projeto, para identificar projetos com funcionários de diferentes departamentos.
 2. Liste os nomes dos projetos que são compartilhados entre departamentos.
 3. Para cada projeto compartilhado, liste todos os funcionários e seus respectivos departamentos.
-

Exercício 6: Análise de Funcionários por Projeto e Departamento (1,0 Pontos)

Instrução: Realize uma análise de quais funcionários de diferentes departamentos trabalham nos mesmos projetos, estabelecendo relações entre eles.

1. Crie um predicado ColegaDeProjeto(X, Y), onde X e Y são funcionários de departamentos diferentes que trabalham no mesmo projeto.
 2. Liste os pares de funcionários que são colegas de projeto e seus respectivos projetos.
 3. Adicione uma contagem de quantos colegas de projeto cada funcionário possui.
-

Boa Prova!