

1. Descrição

Este problema envolve a construção do programa **Academico.lsp** que permitirá associar alunos e professores a disciplinas e a partir dessas associações responder a algumas questões. As funções implementar são as seguintes:

(DEFUN MATRICULAR (ALUNOS DISCIPLINAS TURMA BD) ...)

Argumentos: ALUNOS: Lista contendo nomes de alunos;
DISCIPLINA: Lista com nomes de disciplinas;
TURMA: turma em que os alunos serão matriculados
(apenas uma)
BD: Lista contendo a base de dados.

Retorna: Nova lista contendo a BD atualizada

Descrição: Matricula cada um dos alunos na lista ALUNOS em todas as disciplinas da lista DISCIPLINAS, na TURMA indicada.

(DEFUN CANCELAR-MATRÍCULA (ALUNOS DISCIPLINAS TURMA BD) ...)

Argumentos: ...

Retorna: Nova lista contendo a BD atualizada

Descrição: Cancelar a matrícula cada um dos alunos na lista ALUNOS de todas as disciplinas da lista DISCIPLINAS na TURMA indicada. Disciplinas não vinculadas a nenhum professor e sem alunos matriculados devem ser removidas da base.

(DEFUN VINCULAR (PROFESSORES DISCIPLINAS BD) ...)

Argumentos: ...

Descrição: Vincular cada um dos professores na lista PROFESSORES a cada uma das disciplinas na lista DISCIPLINAS;

Retorna: Nova lista contendo a BD atualizada

Observações: Um professor vinculado a um disciplina está apto a ministrá-la.

(DEFUN REMOVER-VINCULO (PROFESSORES DISCIPLINAS BD) ...)

Argumentos: ...

Descrição: Remover o vínculo de cada um dos professores na lista PROFESSORES a cada uma das disciplinas na lista DISCIPLINAS; Disciplinas não vinculadas a nenhum professor e sem alunos matriculados devem ser removidas da base.

Retorna: Nova lista contendo a BD atualizada

(DEFUN ALUNOS? (BD) ...)

Argumentos: .

Retorna: Lista contendo o nome de todos os alunos cadastrados.

Observação: Um aluno é cadastrado quando é matriculado em alguma disciplina. Alunos não vinculados a nenhuma disciplina devem ser removidos da base.

(DEFUN PROFESSORES? (BD) ...)

Argumentos: .

Retorna: Lista contendo o nome de todos os professores cadastrados.

Observação: Um professor é cadastrado quando é vinculado a alguma disciplina. Professores não vinculados a nenhuma disciplina devem ser removidos da base.

(DEFUN DISCIPLINAS? (BD) ...)

Argumentos: .

Retorna: Lista contendo o nome de todas as disciplinas cadastradas.

Observação: Uma disciplina é cadastrada quando é associada a algum aluno ou professor. Disciplinas não vinculados a nenhum professor ou aluno devem ser removidas da base.

(DEFUN MATRICULADOS? (DISCIPLINA TURMA BD) ...)

Argumentos: .

Retorna: Lista contendo o nome de todos os alunos matriculados na disciplina DISCIPLINA em uma determinada TURMA.

(DEFUN VINCULADOS? (DISCIPLINA BD) ...)

Argumentos: .

Retorna: Lista contendo o nome de todos os professores vinculados à disciplina DISCIPLINA.

(DEFUN CURSA? (ALUNO BD) ...)

Argumentos: .

Retorna: Lista contendo o nome de todas as disciplinas cursadas pelo aluno ALUNO.

(DEFUN MINISTRA? (PROFESSOR BD) ...)

Argumentos: .

Retorna: Lista contendo o nome de todas as disciplinas ministradas pelo professor PROFESSOR.

1.1 Exemplos

Supondo que o nosso programa chame-se *Academico.lsp* e que tenhamos criado o arquivo *BD1.lsp*, mostrado abaixo:

```
:: BD1.lsp
```

```
:: cria uma versão inicial da BD acadêmica.
```

```
(SETQ BD1 'NIL)
```

```
(SETQ BD1 (MATRICULAR '("João Paulo" "Ana Maria") '(APC "Cálculo I") 1 BD1 ))
```

```
(SETQ BD1 (VINCULAR '("Daniele") '(APC "Linguagens Formais") BD1 ))
```

```
(SETQ BD1 (VINCULAR '("Otávio") '("Cálculo I") BD1 ))
```

Um exemplo de sessão no xisp seria:

```
> (LOAD "Academico.lsp") ;; carrega o programa de manutenção da BD
```

```
> (LOAD "BD1.lsp") ;; carrega uma BD exemplo
```

```
...
```

```
> (DISCIPLINAS? BD1)
```

```
(APC "Cálculo I" "linguagens Formais")
```

```
>(MATRICULADOS? 'APC DBAcad1)
("João Paulo" "Ana Maria")
```

...

1.2 Observações:

- As funções não devem provocar efeitos colaterais, ou seja, **não é permitido usar SET e SETQ** nas suas definições. O único uso permitido para SETQ é o indicado nos exemplos acima.
- Usar apenas as funções e formas funcionais discutidas em sala de aula, ou no laboratório. Se quiser usar alguma outra, converse antes com o professor.

2. Entrega

Prazo: 08/05/2019

O projeto pode ser feito em duplas, e um membro da dupla deve realizar a postagem no AVA , com o seguinte conteúdo:
contendo os seguintes arquivos:

- Academico.lsp: arquivo contendo as funções implementadas
- BD1.lsp: arquivo texto com um BD para teste
- Nomes.txt: arquivo texto contendo o nome e ra de cada integrante.

3. Avaliação

Se Funcionalidade $\geq 5,0$

$$\text{NotaPrograma} = (0,7 * \text{Funcionalidade} + 0,3 * \text{Fonte}) * \text{Apresentação}$$

Senão

$$\text{NotaPrograma} = \text{Funcionalidade} * \text{Apresentação}$$

- Funcionalidade (0..10): considerará o atendimento aos requisitos estabelecidos e a confiabilidade do programa (não "trava").
- Fonte (0..10): considerará, principalmente, a legibilidade do programa e o uso adequado dos recursos da linguagem.
- Apresentação (0..1): Nota que reflete o conhecimento demonstrado pelo elemento do grupo escolhido para apresentar e eventualmente modificar o programa entregue.

4. Requisitos Mínimos:

Para que o sistema seja apresentado será necessário que no mínimo as funcionalidades de matrícula (com turma), vincular e cancelar matrícula (com turma) estejam corretamente implementadas e funcionais.

5. Integridade Acadêmica

Na dúvida do que é considerado cola ou plágio, consulte o professor **antes** de entregar qualquer trabalho.