****

**Lógica Aplicada a Computação**

**Lista de Exercícios – 10**

**Professor: Rosalvo Neto**

**Aluno(a):** Marcos Anderson Pereira de Carvalho

**EQUIPE DE FUTSAL**

O código abaixo representa os dados de atletas de um time de futsal em Prolog. Use as informações e faça as consultas no compilador SWISH.

**%fatos**

atleta(gabriel, goleiro).

atleta(joão, pivô).

atleta(caio, ala\_esquerdo).

atleta(jordan, ala\_direito).

atleta(miguel, ala\_direito).

atleta(felipe, pivô).

atleta(gustavo, goleiro).

atleta(milton, ala\_esquerdo).

atleta(bernardo, fixo).

atleta(felix, fixo).

peso(gabriel, 70).

peso(joão, 60.5).

peso(caio, 69).

peso(jordan, 63).

peso(miguel, 61.7).

peso(felipe, 64.2).

peso(gustavo, 67.1).

peso(milton, 72).

peso(bernardo, 68.1).

peso(felix, 70).

idade(gabriel, 21).

idade(joão, 20).

idade(caio, 21).

idade(jordan, 19).

idade(miguel, 21).

idade(felipe, 21).

idade(gustavo, 23).

idade(milton, 18).

idade(bernardo, 20).

idade(felix, 21).

habilidade(gabriel, agilidade).

habilidade(joão, chute).

habilidade(caio, passe).

habilidade(jordan, drible).

habilidade(miguel, velocidade).

habilidade(felipe, chute).

habilidade(gustavo, flexibilidade).

habilidade(milton, drible).

habilidade(bernardo, passe).

habilidade(felix, passe).

1

****

**Lógica Aplicada a Computação**

**Lista de Exercícios – 10**

**Professor: Rosalvo Neto**

**Aluno(a):**

1. **Liste todos os jogadores.**
2. **Quais são os atletas que têm a habilidade de drible?**
3. **Monte uma escalação de atletas com 21 anos.**
4. **Calcule a média de peso de todos os atletas.**
5. **Imagine que você tenha que montar uma escalação ofensiva (Goleiro: ágil, Pivô: chute, Ala esquerdo: drible, Ala direito: velocidade, Fixo: passe). Faça uma regra que retorne os possíveis times de atletas com tais habilidades.**

**RESPOSTAS:**

1. **Liste todos os jogadores.**

**findall(Jogador, atleta(Jogador, \_), ListaJogadores).**

1. **Quais são os atletas que têm a habilidade de drible?**

**findall(Atleta, habilidade(Atleta, drible), ListaAtletas).**

1. **Monte uma escalação de atletas com 21 anos**

timeSub21(sub, Time) :-

atleta(Goleiro, goleiro), idade(Goleiro, Y),

atleta(Pivo, pivô), idade(Pivo, Y),

atleta(AlaEsquerdo, ala\_esquerdo), idade(AlaEsquerdo, Y),

atleta(AlaDireito, ala\_direito), idade(AlaDireito, Y),

atleta(Fixo, fixo), idade(Fixo, Y), Y == 21,

Time = [Goleiro, Pivo, AlaEsquerdo, AlaDireito, Fixo].

**timeSub21(sub, Time).**

**4) Calcule a média de peso de todos os atletas.**

% Base case: A média de peso de uma lista vazia é 0.

media\_peso([], 0)

% Regra recursiva: Calcula a média de peso.

media\_peso([Atleta|Resto], Media) :-

peso(Atleta, Peso),

media\_peso(Resto, MediaResto),

length([Atleta|Resto], NumAtletas),

Media is (MediaResto \* (NumAtletas - 1) + Peso) / NumAtletas

**?- media\_peso([gabriel, joão, caio, jordan, miguel, felipe, gustavo, milton, bernardo, felix], Media).**

**5) Monte uma escalação que seja ofensiva.**

escalação(ofensiva, Time) :-

atleta(Goleiro, goleiro), habilidade(Goleiro, agilidade),

atleta(Pivo, pivô), habilidade(Pivo, chute),

atleta(AlaEsquerdo, ala\_esquerdo), habilidade(AlaEsquerdo, drible),

atleta(AlaDireito, ala\_direito), habilidade(AlaDireito, velocidade),

atleta(Fixo, fixo), habilidade(Fixo, passe),

Time = [Goleiro, Pivo, AlaEsquerdo, AlaDireito, Fixo].

**escalação(ofensiva, Time).**