

1ª Lista de Exercícios de IA 2021.1

Prof. Evandro Costa

Aluno: João Vitor Santos Tavares

- 1) Considerando a **base de conhecimento (Fato + Regras)** com o fato **S = Sim**, na **Memória de Trabalho (MT)** e com as 5 regras seguintes, tal como descrita abaixo, mostre o funcionamento de um engenho de inferência, realizando cada uma das duas estratégias de encadeamento de regras: (i) **encadeamento para frente (com busca em largura)** e (ii) **encadeamento para trás (com busca em profundidade)**, respondendo à pergunta **R = Sim**, mostrando os respectivos grafos de busca que representam as soluções. Mostre um passo a passo da execução da solução no grafo, ressaltando o que foi produzido a cada passo.

R1: **SE** P = Sim **E** Q = Sim **ENTÃO** R = Sim

R2: **SE** S = Sim **ENTÃO** T = Sim

R3: **SE** T = Sim **ENTÃO** P = Sim

R4: **SE** S = Sim **ENTÃO** V = Sim

R5: **SE** V = Sim **ENTÃO** Q = Sim

MT = { S = Sim }

R:

Encadeamento pra frente:

Busca adjacência = S(SIM);

DERIVAÇÕES POSSÍVEIS = (S) = R2, R4;

R NÃO ENCONTRADO = Continuar;

Busca na adjacência = T(SIM), V(SIM);

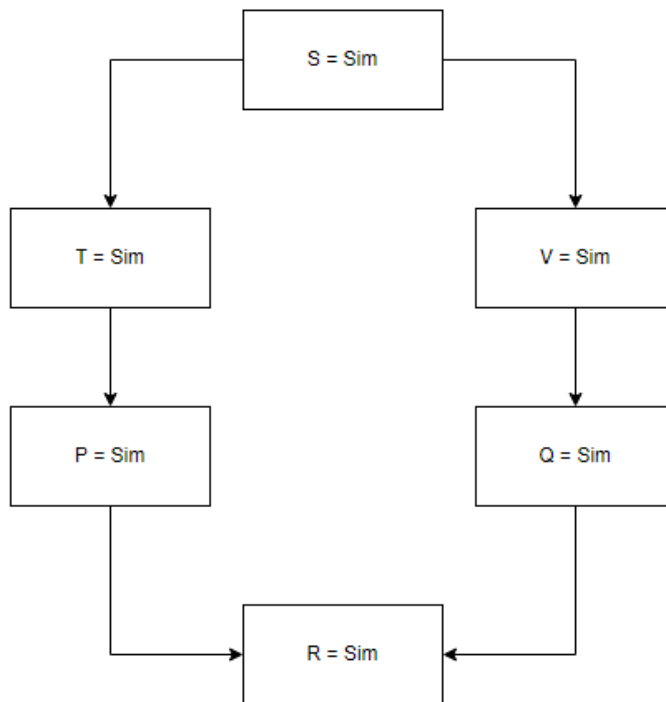
DERIVAÇÕES POSSÍVEIS = (T, V) = R3, R5;

R NÃO ENCONTRADO = Continuar;

Busca na adjacência = P(SIM), Q(SIM);

DERIVAÇÕES POSSÍVEIS = (P, Q) = R;

R ENCONTRADO = return (S)



Provar R, condição: $(P \wedge Q)$ em R1;

Provar Q, condição: V em R5;

Provar V, condição: S em R4;

S = SIM, return;

V = SIM, return;

Q = SIM, return;

Provar P, condição: T em R3;

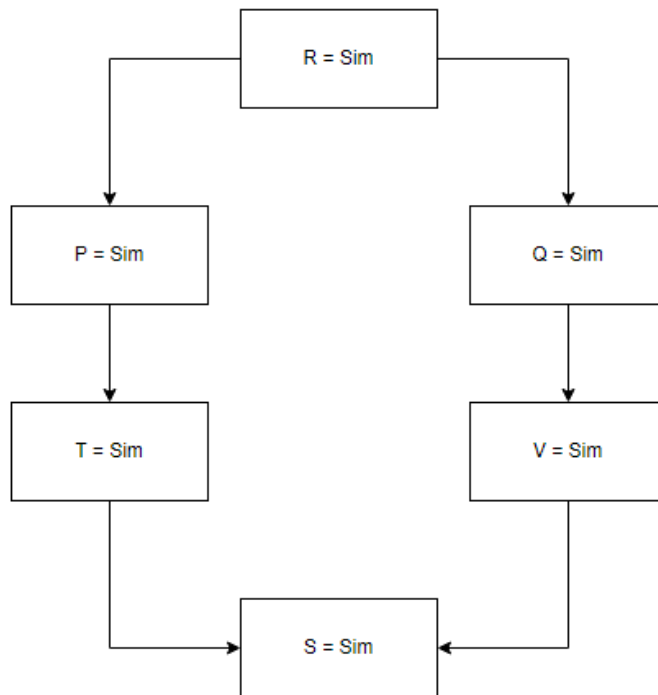
Provar T, condição: S em R2;

S = SIM, return;

T = SIM, return;

P = SIM, return;

R = SIM, return;



2) Apresente um **algoritmo** e **implemente** uma máquina (engenho) de inferência que realize:

2.1) uma estratégia de encadeamento para frente;

2.2) uma estratégia de encadeamento para trás;

2.3) uma estratégia de encadeamento misto (para frente e para trás).

Defina uma linguagem para representação de conhecimento para descrever base de regras e fatos, considerando um formato seguindo o exemplo a seguir, a ser mais detalhado pelo professor da disciplina, durante a aula. com uma aplicação que demande na solução a realização de uma estratégia de encadeamento misto, bem como mostre a execução do algoritmo apresentado sobre esse exemplo. Para auxiliar no entendimento da questão e avaliação, considere o exemplo hipotético a seguir (adaptado do Livro Inteligência Artificial, do Ben Coppin), ilustrando uma possível demanda por encadeamento misto.

R1: SE Dor de cabeça = Sim ENTÃO Receitar analgésico

R2: SE Dor de cabeça = Sim E Garganta inflamada = Sim E Tosse = Sim

ENTÃO Diagnóstico = Gripe

R3: SE Cansaço = Sim E Dor de cabeça = Sim

ENTÃO Diagnóstico = Mononucleose infecciosa

R4: SE Cansaço = Sim E Garganta inflamada = Sim

ENTÃO Diagnóstico = Amigdalite

R5: SE Cansaço = Sim ENTÃO Diagnóstico = Estresse

Neste caso, considere uma memória de trabalho com um valor inicial Dor de Cabeça = Sim, supondo uma situação com propósito de diagnóstico na qual um determinado usuário informa ao sistema que está com dor de cabeça. Assim, a máquina de inferência pode iniciar com encadeamento para frente para derivar novos fatos, após isso pode fazer perguntas para prosseguir sua análise usando encadeamento para trás.

Outras bases de conhecimento que devem ser usadas como teste:

Regra1: Se Rendimento = Alto então conceda_empréstimo = Sim. Regra2: Se Rendimento = Médio E É_bacharel_ou_superior = Sim E Tem_Emprego = Sim então conceda_empréstimo = Sim.

Regra 3: Se Rendimento = Médio E É_bacharel_ou_superior = Sim E Tem_Emprego = Não então continua_a_investigar = Sim.

Regra 4: Se Rendimento = Médio E É_bacharel_ou_superior = Não E Tem_Emprego = Não então conceda_empréstimo = Não.

Regra 5: Se Rendimento = Médio E É_bacharel_ou_superior = Não E Tem_Emprego = Sim então continua_a_investigar = Sim.

Regra 6: Se Rendimento = Baixo E Referências = Boas então continua_a_investigar = Sim.

Regra 7: Se Rendimento = Baixo E Referências = Más então conceda_empréstimo = Não.

R1: IF a person has a risk of heart attack

AND a person has had a previous heart attack

THEN give that person digitals

R2: IF a person has left quadratic pain

AND a person has high blood pressure

THEN that person has a risk of heart attack

R3: IF a person has raised intraocular pressure

THEN that person has high blood pressure

- Suponha MT inicial = {A, B, C, E, G, H}
- Adaptado do Livro editado por R. Forsyth, 149

Considere os seguintes fatos:

Fact A Smith has raised intraocular pressure

Fact B a person has had a previous heart attack

Fact C Smith has left quadratic pain

Fact D Smith has high blood pressure

Fact E Smith is a heavy smoker

Fact F Smith has a risk of heart attack
Fact G Jones is an asthmatic
Fact H Jones is a heavy smoker
Fact I Give Smith digitals

Adaptado do Livro editado por R. Forsyth, 149

R1: IF x tem F AND x tem B
 THEN x recebe I
R2: IF x tem C AND x tem D
 THEN x tem F
R3: IF x tem A
 THEN x tem D

- Suponha MT inicial = {A, B, C, E, G, H}
- Adaptado do Livro editado por R. Forsyth, 149

R:

pra frente:

```
Ozzy = False
Distortion = False
Tony = False
Black_sabbath = False
Bill = False
Geezer = False
Politics_lyrics = True

def factual_base():
    global Ozzy
    Ozzy = True

if __name__ == '__main__':
    factual_base()
    c = 0
    while c < 10:

        if Black_sabbath:
            print(f'Black_sabbath = True')
            break

        if Politics_lyrics and Distortion:
            Black_sabbath = True

        if Ozzy:
            Tony = True
            Geezer = True
```

```

    if Ozzy and Geezer:
        Bill = True

    if Ozzy and Tony:
        Distortion = True
        Politics_lyrics = True

else:
    print(f'Black_sabbath = False')

```

pra trás:

```

Ozzy = False
Distortion = False
Tony = False
Black_sabbath = False
Bill = False
Geezer = False
Politics_lyrics = True

def factual_base():
    global Ozzy
    Ozzy = True

if __name__ == '__main__':
    factual_base()
    c = 0
    while c < 10:

        if Ozzy and Tony:
            Distortion = True
            Politics_lyrics = True

        if Ozzy and Geezer:
            Bill = True

```

```

    if Ozzy:
        Tony = True
        Geezer = True

    if Politics_lyrics and Distortion:
        Black_sabbath = True

    if Black_sabbath:
        print(f'Black_sabbath = True')
        break

else:
    print(f'Black_sabbath = False')

```

Ambos os lados:

```

Ozzy = False
Distortion = False
Tony = False
Black_sabbath = False
Bill = False
Geezer = False
Politics_lyrics = True

def factual_base():
    global Ozzy
    Ozzy = True

if __name__ == '__main__':
    factual_base()
    c = 0
    while c < 10:
        if Ozzy and Geezer:
            Bill = True

        if Ozzy and Tony:
            Distortion = True
            Politics_lyrics = True

        if Black_sabbath:
            print(f'Black_sabbath = True')
            break

```

```

if Politics_lyrics and Distortion:
    Black_sabbath = True

if Ozzy:
    Tony = True
    Geezer = True

else:
    print(f'Black_sabbath = False')

```

- 3) Considere a base de conhecimento da questão 2, ampliando-a com acréscimo de mais 5 regras (hipotéticas) para outros diagnósticos, por exemplo, de resfriado, dengue, covid19. Após isso, use a ferramenta que você desenvolveu na questão anterior e também use uma ferramenta de sistema baseado em conhecimento qualquer, por exemplo, Drools ou JESS ou Expert Sinta ou Prolog ou alguma **ferramenta em Python**, para realizar consultas de diagnóstico, observando e descrevendo o funcionamento da máquina de inferência. Compare os resultados.
- 4) Essas versões iniciais de máquina de inferência para agentes baseados em regras, tendo algoritmos com encadeamento de regras, tais como discutidas em aula, mostram-se ineficientes quando se tem bases de conhecimento com muitas regras e que os fatos não mudam com frequência. Para amenizar esse problema, criou-se o algoritmo RETE (o qual tem sido melhorado cada vez mais ao longo dos anos). Assim, pesquise e descreva o propósito do Algoritmo RETE, explicando o seu funcionamento com um exemplo ilustrativo.

R:

O Algoritmo RETE é um algoritmo que se utiliza das bases de conhecimento e regras de produção. A base de conhecimento é utilizada nesse algoritmo para consultar as regras de produção e, a partir destas, tomar decisões. Esse algoritmo também se utiliza de um sistema de nós para gerar retorno de resultados.

O RETE é ideal em cenários de negócios e alguns outros cenários complexos. Com a utilização de variáveis e avaliação tendo como base as regras da base, o RETE consegue extrair muitos detalhes sobre o que está ocorrendo em um dado cenário. O mesmo pode ser implementado de diversas formas, bem como por linguagens diferentes. Exemplos práticos de sua aplicação é uma plataforma em Java chamada Drools.

