

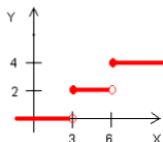
Inteligência Artificial

Aula 4 - vídeo 1 - Cut

26 de agosto de 2020

Cut

Defina a seguinte função em Prolog:



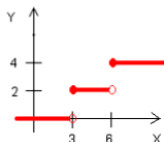
Regra 1 : se $X < 3$ então $Y = 0$.

Regra 2 : se $3 \leq X < 6$ então $Y = 2$.

Regra 3 : se $X \geq 6$ então $Y = 4$.

Cut

Defina a seguinte função em Prolog:



Regra 1 : se $X < 3$ então $Y = 0$.

Regra 2 : se $3 \leq X < 6$ então $Y = 2$.

Regra 3 : se $X \geq 6$ então $Y = 4$.

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$

Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$

?- $f(1, Y), 2 < Y.$

Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$

?- $f(1, Y)$, $2 < Y.$

Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$

?- $f(1, Y)$, $2 < Y.$

1



Cut

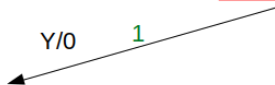
Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$

?- $f(1, Y)$, $2 < Y.$



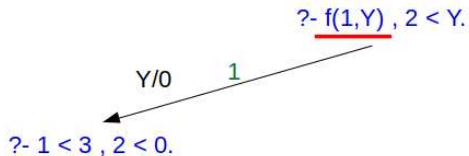
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



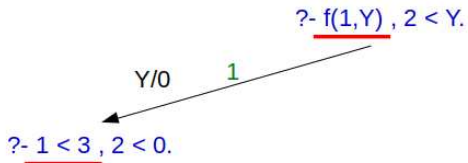
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



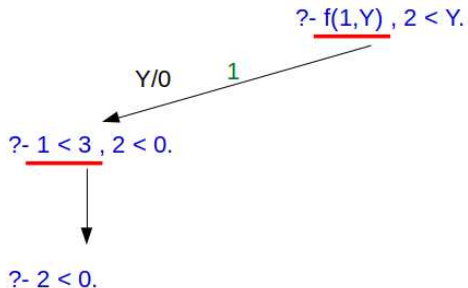
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



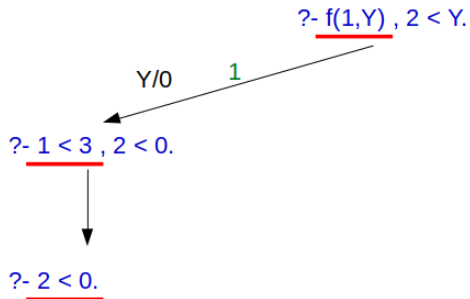
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



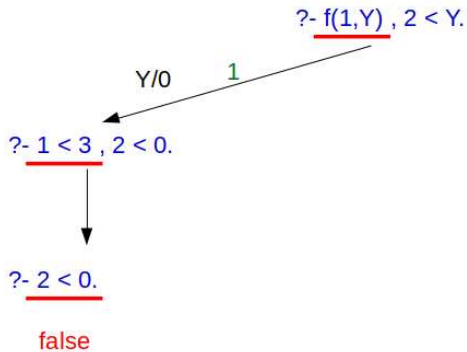
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



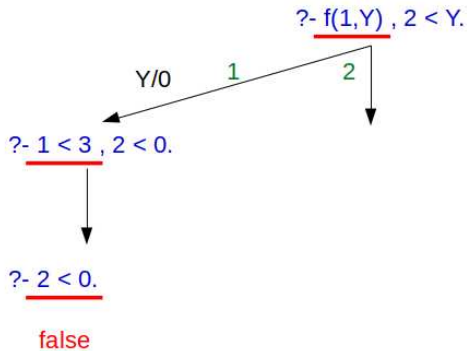
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



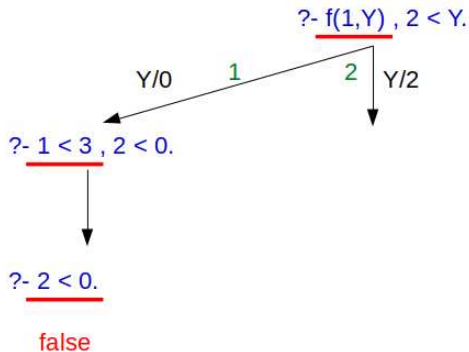
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



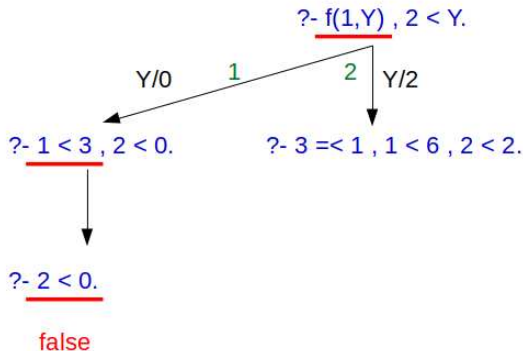
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



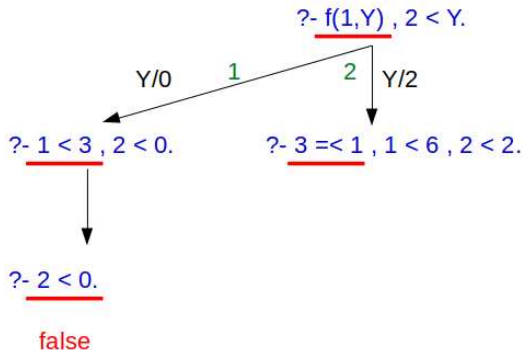
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



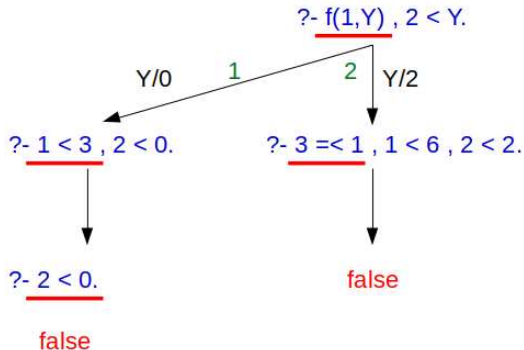
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



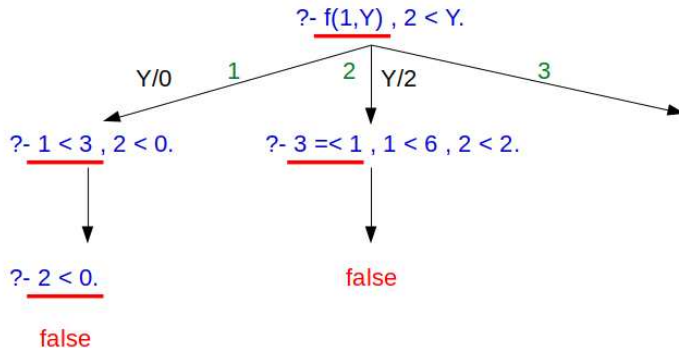
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



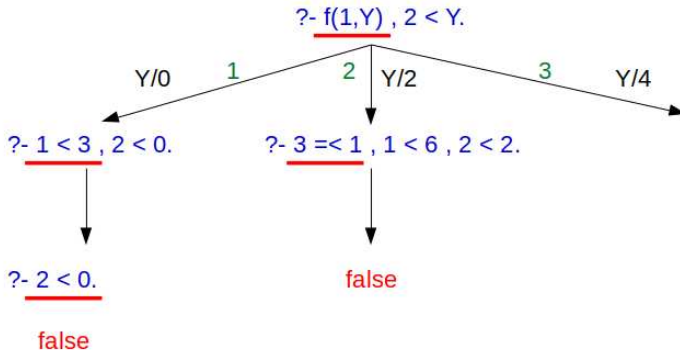
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



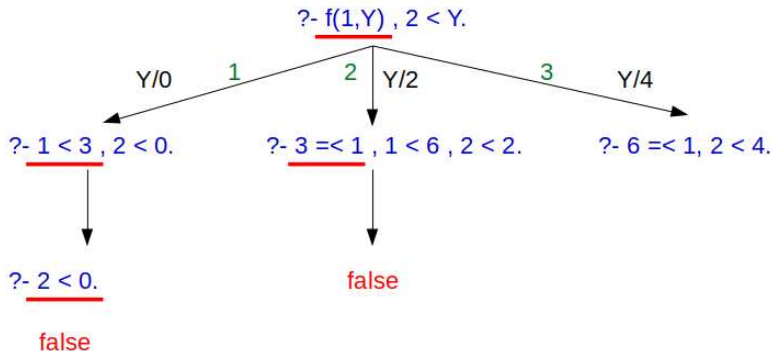
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



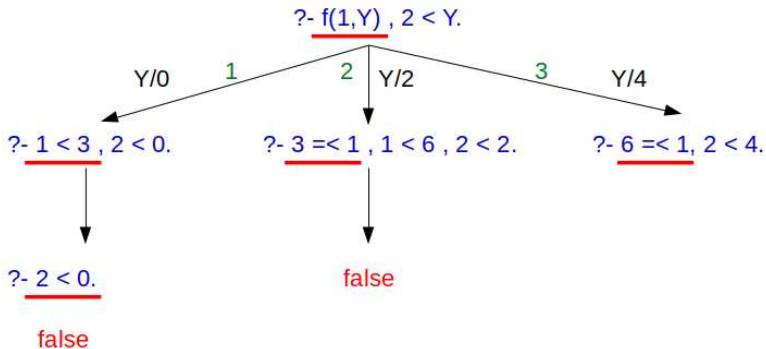
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



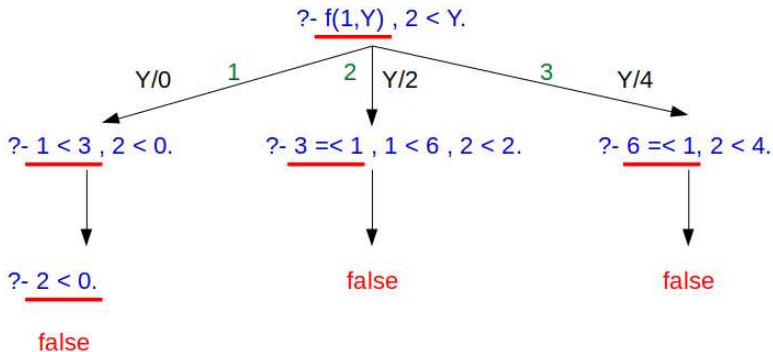
Cut

Programa

$f(X, 0) :- X < 3.$

$f(X, 2) :- 3 \leq X, X < 6.$

$f(X, 4) :- 6 \leq X.$



Cut

Faz uma poda da árvore de derivação a partir do ponto em que ! aparece.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- 3 \leq X, X < 6, !.$

$f(X, 4) :- 6 \leq X.$

?- $f(1, Y), 2 < Y.$

Cut

Faz uma poda da árvore de derivação a partir do ponto em que ! aparece.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- 3 \leq X, X < 6, !.$

$f(X, 4) :- 6 \leq X.$

?- f(1,Y) , 2 < Y.

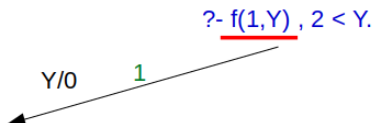
Cut

Faz uma poda da árvore de derivação a partir do ponto em que ! aparece.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- 3 \leq X, X < 6, !.$

$f(X, 4) :- 6 \leq X.$



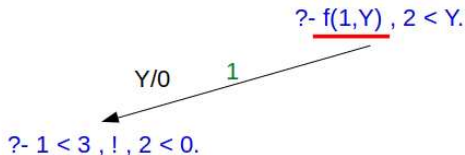
Cut

Faz uma poda da árvore de derivação a partir do ponto em que ! aparece.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- 3 \leq X, X < 6, !.$

$f(X, 4) :- 6 \leq X.$



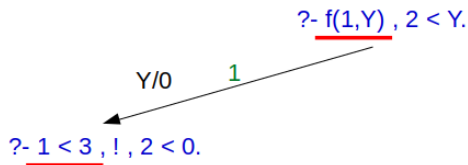
Cut

Faz uma poda da árvore de derivação a partir do ponto em que ! aparece.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- 3 \leq X, X < 6, !.$

$f(X, 4) :- 6 \leq X.$



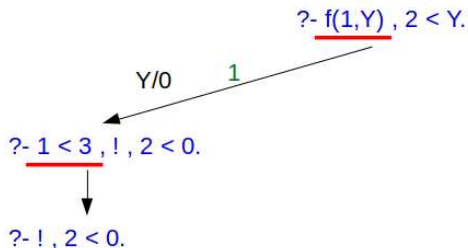
Cut

Faz uma poda da árvore de derivação a partir do ponto em que ! aparece.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- 3 \leq X, X < 6, !.$

$f(X, 4) :- 6 \leq X.$



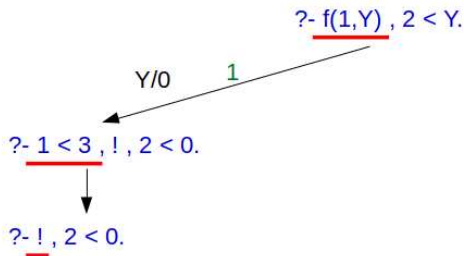
Cut

Faz uma poda da árvore de derivação a partir do ponto em que ! aparece.

$f(X, 0) \text{ :- } X < 3, !.$

$f(X, 2) \text{ :- } 3 \leq X, X < 6, !.$

$f(X, 4) \text{ :- } 6 \leq X.$



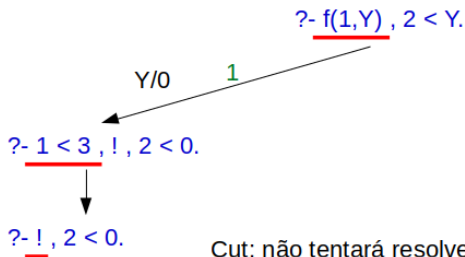
Cut

Faz uma poda da árvore de derivação a partir do ponto em que **!** aparece.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- 3 \leq X, X < 6, !.$

$f(X, 4) :- 6 \leq X.$



Cut: não tentará resolver usando as outras cláusulas do programa. Com isso, os outros ramos não serão gerados.

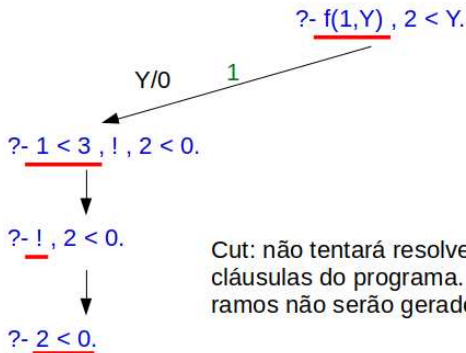
Cut

Faz uma poda da árvore de derivação a partir do ponto em que **!** aparece.

$f(X, 0) \text{ :- } X < 3, \text{!}.$

$f(X, 2) \text{ :- } 3 \leq X, X < 6, \text{!}.$

$f(X, 4) \text{ :- } 6 \leq X.$



Cut: não tentará resolver usando as outras cláusulas do programa. Com isso, os outros ramos não serão gerados.

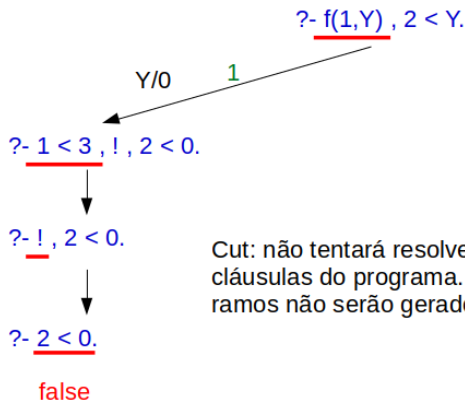
Cut

Faz uma poda da árvore de derivação a partir do ponto em que **!** aparece.

$f(X, 0) \text{ :- } X < 3, \text{!}.$

$f(X, 2) \text{ :- } 3 \leq X, X < 6, \text{!}.$

$f(X, 4) \text{ :- } 6 \leq X.$



Cut: não tentará resolver usando as outras cláusulas do programa. Com isso, os outros ramos não serão gerados.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- 3 \leq X, X < 6, !.$

$f(X, 4) :- 6 \leq X.$

Consulta : $? - f(7, Y).$

- **Regra 1 :** $7 < 3$ falha. (! não alcançado)
- **Regra 2 :** $3 \leq 7$ sucesso, mas $7 < 6$ falha.(! não alcançado)
- **Regra 3 :** $6 \leq 7$ sucesso.

Resposta: $Y = 4.$

Cut

$f(X, 0) :- X < 3, !.$
 $f(X, 2) :- X < 6, !.$
 $f(X, 4).$

$f(X, 0) :- X < 3.$
 $f(X, 2) :- X < 6.$
 $f(X, 4).$

Consulta : $? - f(1, Y).$

● Programa com cut

- Casa $f(1, Y)$ com a cabeça da regra 1 ($f(X, 0)$) : $X = 1, Y = 0.$
- $(1 < 3)$ é satisfeito e o cut é atingido.
- Resposta $Y = 0.$
- Nenhum outro casamento é tentado.

$f(X, 0) :- X < 3, !.$

$f(X, 2) :- X < 6, !.$

$f(X, 4).$

$f(X, 0) :- X < 3.$

$f(X, 2) :- X < 6.$

$f(X, 4).$

Consulta : $? - f(1, Y).$

● Programa sem cut

- Casa $f(1, Y)$ com a cabeça da regra 1 ($f(X, 0)$) : $X = 1, Y = 0$.
- $(1 < 3)$ é satisfeito e o cut é atingido. **Resposta:** $Y = 0$.
- Casa $f(1, Y)$ com a cabeça da regra 2 ($f(X, 2)$) : $X = 1, Y = 2$.
- $(2 < 6)$ é satisfeito. **Resposta** $Y = 2$.
- Casa $f(1, Y)$ com a cabeça da regra 3 ($f(X, 4)$) : $X = 1, Y = 4$. **Resposta:** $Y = 4$.
- Não temos mais regras/fatos com que casar $f(1, Y)$. **Resposta:** *false*.

Objetivo G 'casa' com H onde :

$$H : -B_1, B_2, \dots, B_m, !, \dots, B_n.$$

- **Encontra !:** O sistema já encontrou alguma solução para B_1, B_2, \dots, B_m .
- **Executa !:** As soluções correntes para B_1, B_2, \dots, B_m ficam fixas e todas as alternativas são descartadas.
Outras cláusulas que 'casam' com G também são descartadas.

Inteligência Artificial

Aula 4 - vídeo 1 - Cut

26 de agosto de 2020