

Retrocesso e corte

Marco A L Barbosa

malbarbo.pro.br

Os exercícios sem referências estão licenciados com uma Licença Creative Commons - Atribuição-CompartilhaIgual 4.0 Internacional.



<https://github.com/malbarbo/na-proglog>

- 1) [pip 7.9] Três números naturais A, B, C são chamados de terno Pitagórico se $A^2 + B^2 = C^2$. Projete um predicado `terno_pitagorico(A, B, C)` que gere ternos Pitagóricos.

```
?- terno_pitagorico(A, B, C).  
A = 3,  
B = 4,  
C = 5 ;  
A = 6,  
B = 8,  
C = 10 ;  
A = 5,  
B = 12,  
C = 15 ;  
...
```

- 2) [pp99 1.26] Defina um predicado `combinacao(K, L, C)` que é verdadeiro se C é uma combinação de K elementos de L . Este predicado deve ser capaz de gerar todas as combinações de K elementos via retrocesso. Exemplo

```
?- combinacao(3, [a, b, c, d], C).  
C = [a, b, c] ;  
C = [a, b, d] ;  
C = [a, c, d] ;  
C = [b, c, d] ;  
false.
```

- 3) Defina um predicado `subconjunto(L, S)` que é verdadeiro se S é um subconjunto de L . Este predicado deve ser capaz de gerar todos os subconjuntos de L via retrocesso. Exemplo

```
?- subconjunto([a, b, c], S).  
S = [] ;  
S = [a] ;  
S = [b] ;  
S = [c] ;  
S = [a, b] ;  
S = [a, c] ;  
S = [b, c] ;  
S = [a, b, c] ;  
false.
```

- 4) Dado um conjunto de números inteiros e um inteiro S , o problema da soma dos subconjuntos consiste em verificar se existe um subconjunto não vazio cuja soma é S . Defina um predicado `soma_subconjunto(A, S, P)` que é verdadeiro se P é um subconjunto de A e a soma dos elementos de P é S . Uma estratégia simples (e ingênua) para implementar este predicado é testar os subconjunto até encontrar um que tenha a soma esperada.

```
?- soma_subconjunto([-7, -3, -2, 5, 8], 0, P).
```

$P = [-3, -2, 5]$.

- 5) [pip 11-8] Implemente um predicado para encontrar todas as formas de posicionar 4 rainhas em um tabuleiro de xadrez 4x4 de maneira que nenhuma rainha ataque outra. Uma forma de fazer este predicado é criar um gerador de permutações e testar se as rainhas foram posicionadas de maneira correta.
- 6) Quatro crianças estão visitando o aquário pela primeira vez. Dado os seguintes detalhes
- Nem a criança que quer ver o peixe palhaço nem a que quer ver o tubarão levou bolacha.
 - Quem levou sanduíche é 1 ano mais novo que a criança que quer ver o peixe palhaço.
 - A criança que levou fruta é 1 ano mais nova do que quem deseja ver o tubarão.
 - Danilo é 2 anos mais velho do que a criança que levou fruta.
 - Alan não tem 6 anos.
 - A criança que levou salgadinho quer ver o tubarão.
 - Quem deseja ver o tubarão é o Renan ou o menino de 9 anos.
 - Quem levou sanduíche tem 8 anos ou quer ver o leão-marinho.

Projete um predicado `aquario(?C1, ?C2, ?C3, ?C4)`, que é verdadeiro se `C1`, `C2`, `C3` e `C4` são termos `crianca(Nome, Idade, Animal, Lanche)` que atendem as restrições anteriores.

Referências

- [lpn]. [Lear Prolog Now](#)
- [pip]. Programming in Prolog.
- [pp99]. [99 problemas para resolver em \(Prolog\)](#)