10.10

:-use_module(library(clpfd)).

generate & test
vs
constraint & generate

1. Declarar Variáveis

length(Vars, 10)

X < Z,
A in 1..5, B in 1..5,
A #< B,

2. Declarar Domínios

SingleVar in 1..5
domain(Vars, 1, 10)

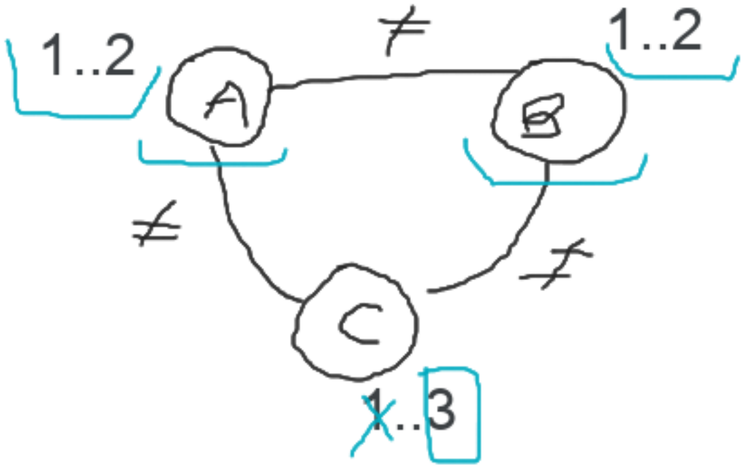
A in 1..4
B in 2..5

3. Declarar Restrições

# #= #\= #< #> #=< #>=
all_different(ListOfVars) all_distinct(Vars)

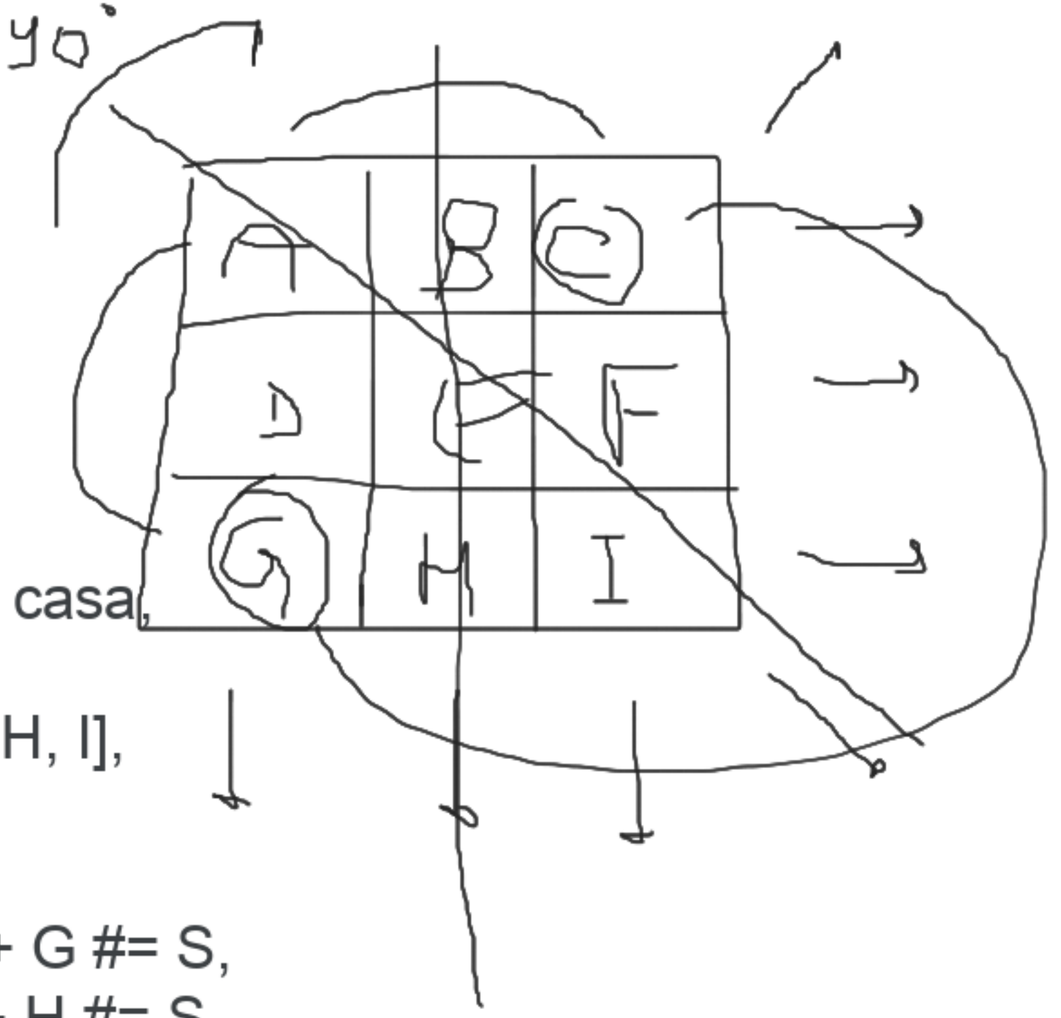
4. Pesquisar Solução

labeling([ ], Vars)



3 Vars diferentes, soma = produto

```
L = [A, B, C],
domain(L, 1, 1000),
all_distinct(L),
A + B + C #= A * B * C,
labeling( [ ], L).
```




```
% Posição representa a casa,
valor é o valor
L = [A, B, C, D, E, F, G, H, I],
domain(L, 1, 9),
all_distinct(L),
A + B + C #= S,  A + D + G #= S,
D + E + F #= S,  B + E + H #= S,
G + H + I #= S,  C + F + I #= S,
A + E + I #= S, C + E + G #= S,
A #< C, A #< G,  C#< G,
labeling([ ], L).
```

6, 1, 7, 4, 5, 3

PUP
PPP

S E N D
+ M O R E

M O N E Y