



INSTITUTO POLITECNICO NACIONAL
ESCUELA SUPERIOR DE COMPUTO



INSTITUTO POLITECNICO NACIONAL
ESCUELA SUPERIOR DE COMPUTO
PARADIGMAS DE PROGRAMACION

Vazquez Perez Angel Martin

Grupo 3BV2

Boleta: 2023630057

Caso 0:implementar funciones en prolog con
generar-comprobar

Introduccion

En esta tareamediante el modelo de generar-comprobar, se buscaran implementar distintos casos donde se quiera saber si la suma de 2 numeros me resulta un par, incluyendo que estos 2 numeros que se esten sumando sean igualmente pares.

FUNCIONES que usaremos dentro de los 10 programas

```
par(X):- 0 == X mod 2 .  
impar(X):- 1 == X mod 2 .  
suma(X,Y,Z):- X + Y == Z.  
resta(Z,X,Y):- Y == Z - X .  
mayor(X,Y):- X > Y.  
menor(X,Y):- X<Y.
```

CASO 1

```
pares_suma_caso1(Z,X,Y):- par(X),           %generar  
    par(Y),                                   %generar  
    suma(X,Y,Z).                             %comprobar
```

PRUEBA CASO 1

```
?- pares_suma_caso1(48,32,16).  
true.
```

```
?- pares_suma_caso1(48,35,13).  
false.
```

CASO 2

```
pares_suma_caso2(Z,X,Y):-menor(X,Z), %comprobar  
    menor(Y,Z),                    %comprobar  
    suma(X,Y,Z),                   %comprobar  
    par(X),                        %comprobar  
    par(Y).                        %comprobar
```

PRUEBA CASO 2

```
?- pares_suma_caso2(36,16,20).  
true.
```

```
?- pares_suma_caso2(36,46,-10).  
false.
```

CASO 3

```
pares_suma_caso3(Z,X,Y):-menor(X,Z), %comprobar
    par(X), %comprobar
    suma(X,Y,Z), %generar
    par(Y). %comprobar
```

PRUEBA CASO 3

```
?- pares_suma_caso3(488,322,166).
true.
```

```
?- pares_suma_caso3(48,22,66).
false.
```

CASO 4

```
pares_suma_caso4(Z,X,Y):-suma(X,Y,Z),%generar
    par(X), %comprobar
    par(Y). %comprobar
```

PRUEBA CASO 4

```
?- pares_suma_caso4(24,16,8).
true.
```

```
?- pares_suma_caso4(24,14,88).
false.
```

CASO 5

```
 pares_suma_caso5(Z,X,Y):- par(Z),%comprobar  
                             suma(X,Y,Z), %generar  
                             par(X).      %comprobar
```

PRUEBA CASO 5

```
?- pares_suma_caso5(44,18,26).  
true.
```

```
?- pares_suma_caso5(33,15,18).  
false.
```

CASO 6

```
 pares_suma_caso6(Z,X,Y):-resta(Z,X,Y), %generar  
                             par(Z),      %comprobar  
                             par(X).      %comprobar
```

PRUEBA CASO 6

```
?- pares_suma_caso6(22,18,4).  
true.
```

```
?- pares_suma_caso6(27,15,12).  
false.
```

CASO 7

```
pares_suma_caso7(Z,X,Y):- not(impar(Z)), %comprobar
                           suma(X,Y,Z),    %generar
                           not(impar(X)).  %comprobar
```

PRUEBA CASO 7

```
?- pares_suma_caso7(58,36,22).
true.
```

```
?- pares_suma_caso7(96,76,21).
false.
```

CASO 8

```
pares_suma_caso8(Z,X,Y):- not(menor(Z,Y)), %comprobar
                           par(Y),          %comprobar
                           not(impar(X)),   %comprobar
                           suma(X,Y,Z).     %generar
```

PRUEBA CASO 8

```
?- pares_suma_caso8(28,26,2).
true.
```

—

```
?- pares_suma_caso8(87,55,32).
false.
```

CASO 9

```
pares_suma_caso9(Z,X,Y):- mayor(Z,Y),           %comprobar
                           resta(Z,X,Y),          %generar
                           not(impar(X)),          %comprobar
                           par(Y).                %comprobar
```

PRUEBA CASO 9

```
?- pares_suma_caso9(118,98,20).
true.
```

```
?- pares_suma_caso9(111,87,24).
false.
```

CASO 10

```
pares_suma_caso10(Z,X,Y):- menor(X,Z),%comprobar
                           suma(X,Y,Z),  %generar
                           par(X),       %comprobar
                           par(Y),       %comprobar
                           resta(Z,X,Y). %generar
```

PRUEBA CASO 10

```
?- pares_suma_caso10(96,12,84).
true.
```

```
?- pares_suma_caso10(9,7,2).  ?- pares_suma_caso10(9,7,2).
false.                       false.
```