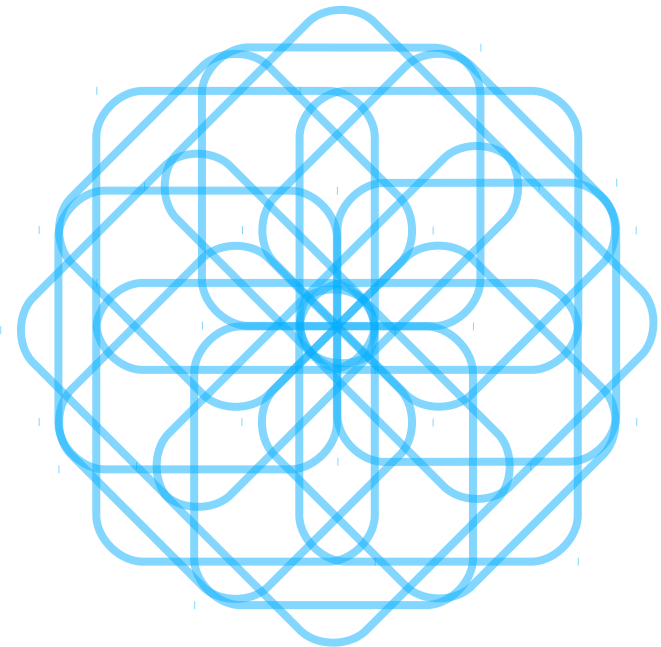
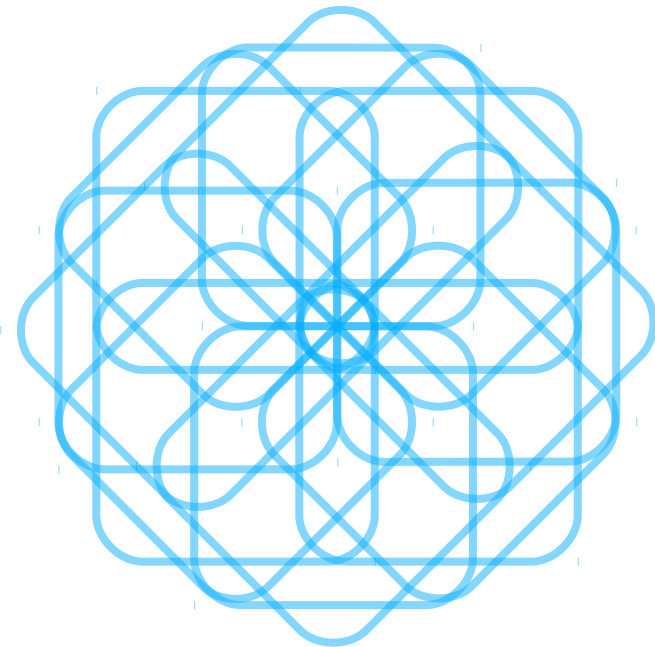


# Algoritmos Genéticos



# Algoritmo Genético Binario

Si se tiene un problema que se desea resolver y se puede representar cada posible solución al problema como una cadena de bits, entonces un Algoritmo Genético puede resolver el problema.



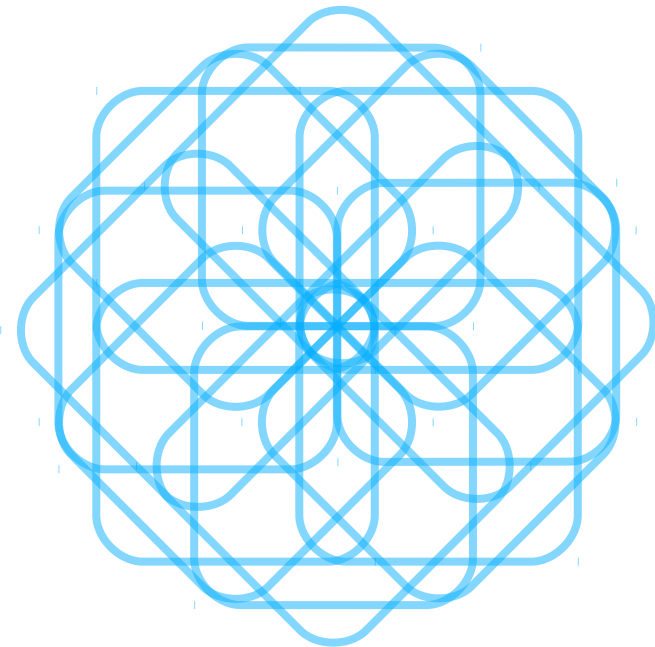
# Representación

Supongase que se tiene que diseñar un robot de bajo peso con suficiente poder para navegar terreno difícil y suficiente energía para no regresar a la base muy seguido.

Parámetros:

Tipo de motor y tamaño

Tipo de fuente de energía y tamaño



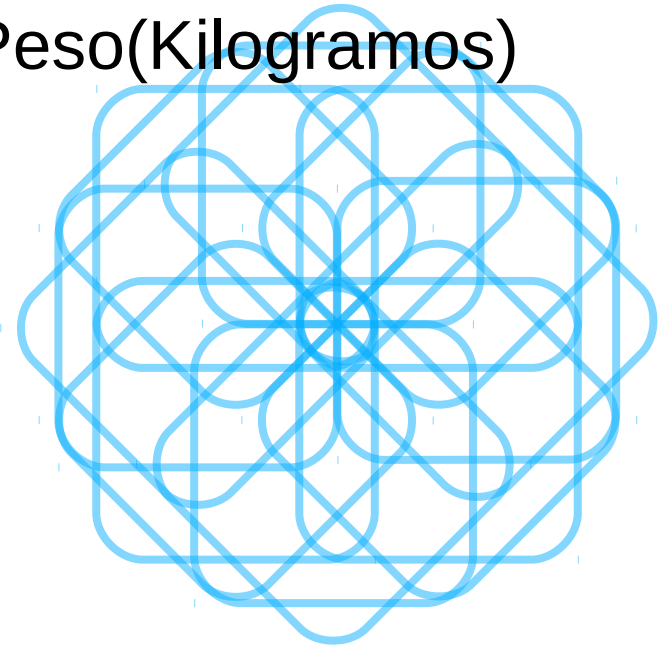
Motor	Fuente de Energía
000 = 5 volts motor a pasos	000 = 12 volts batería nickel-cadmio
001 = 9 volts motor a pasos	001 = 24 volts batería nickel-cadmio
010 = 12 volts motor a pasos	010 = 12 volts batería de litio
011 = 24 volts motor a pasos	011 = 24 volts batería de litio
100 = 5 volts servo motor	100 = 12 volts panel solar
101 = 9 volts servo motor	101 = 24 volts panel solar
110 = 12 volts servo motor	110 = 12 volts panel solar
111 = 24 volts servo motor	111 = 24 volts panel solar

# Fitness

No se puede simular evolución si no se tiene una buena definición de fitness o aptitud

Para el ejemplo se podría plantear una función de aptitud como la siguiente:

Fitness = Energia(horas)+Potencia(Watts)-Peso(Kilogramos)



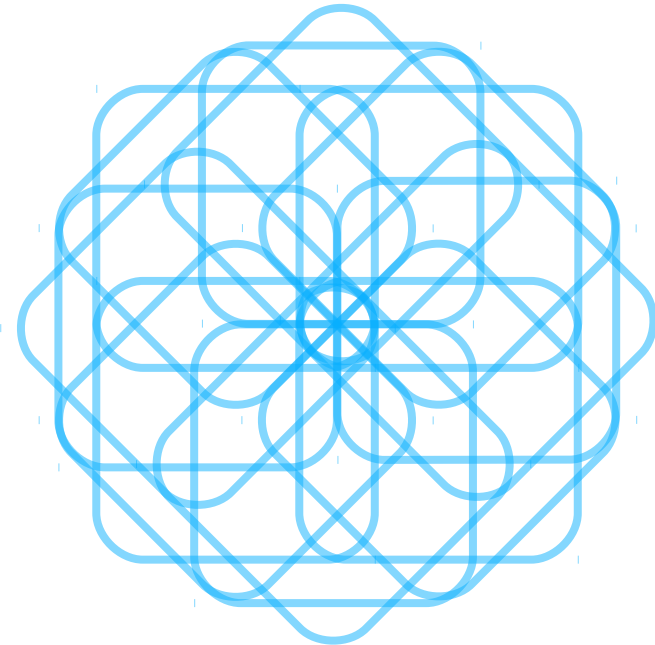
# Individuos

Individuo 1

24 volts motor a pasos 011	24 volts batería nickel-cadmio 001
-------------------------------	---------------------------------------

Individuo 2

5 volts servo motor 100	24 volts panel solar 111
----------------------------	-----------------------------



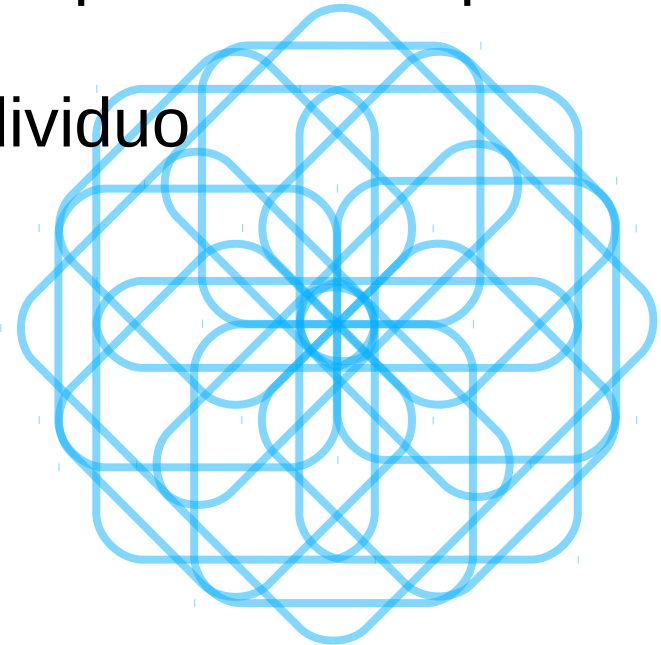
**Alelo:** Cada bit en un individuo

**Gen:** Una secuencia de bits que contenga una sección de información de un individuo

**Genotipo:** Un gen específico

**Fenotipo:** El parámetro específico al problema que representa el genotipo

**Cromosoma:** Colección de genes de un individuo



24 volts motor a pasos 011	24 volts batería nickel-cadmio 001
-------------------------------	---------------------------------------

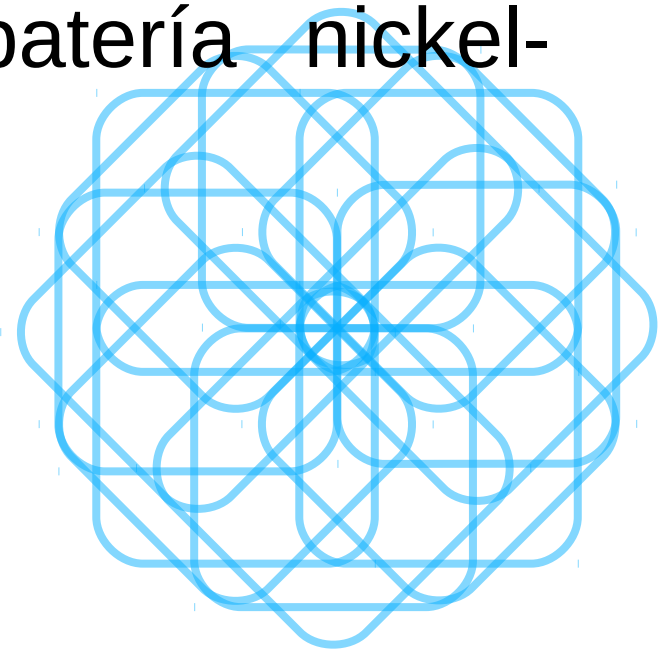
2 Genes, uno para motor y otro para batería

Genotipo de motor: 011

Fenotipo de motor: 24 volts motor a pasos

Genotipo de energía: 001

Fenotipo de batería: 24 volts batería nickel-cadmio

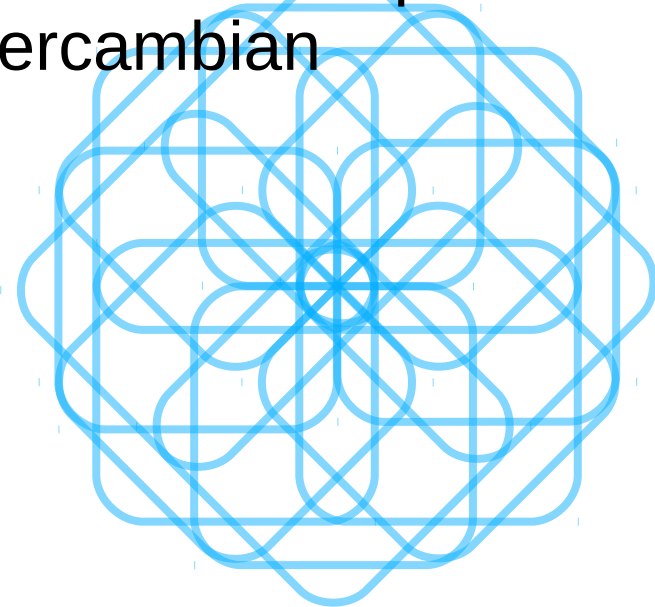




# Selección y Cruza

En terminos de los algoritmos genéticos la *cruza* significa que cada individuo comparte algo de su información con su descendencia.

Para ello se debe seleccionar una posición aleatoria a partir de la cual los alelos de dos individuos se intercambian



# Cruza

Padres

011001

*100111*

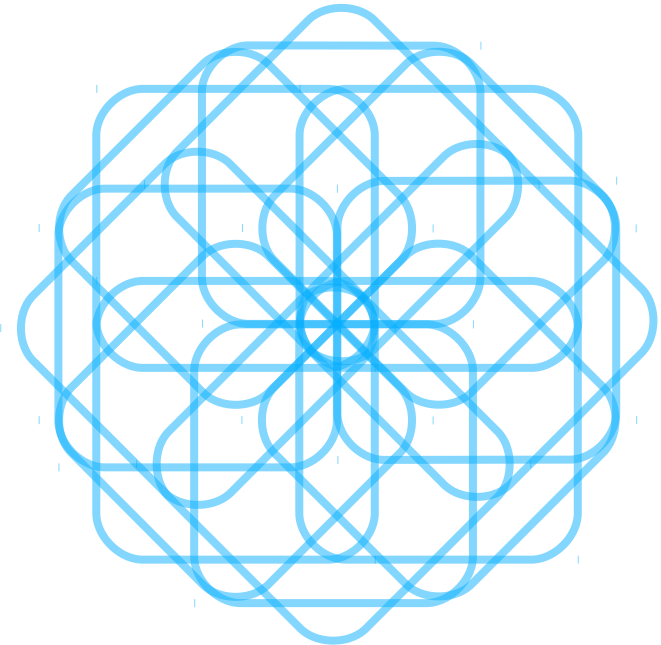


Punto de cruza

Hijos

0110*11*

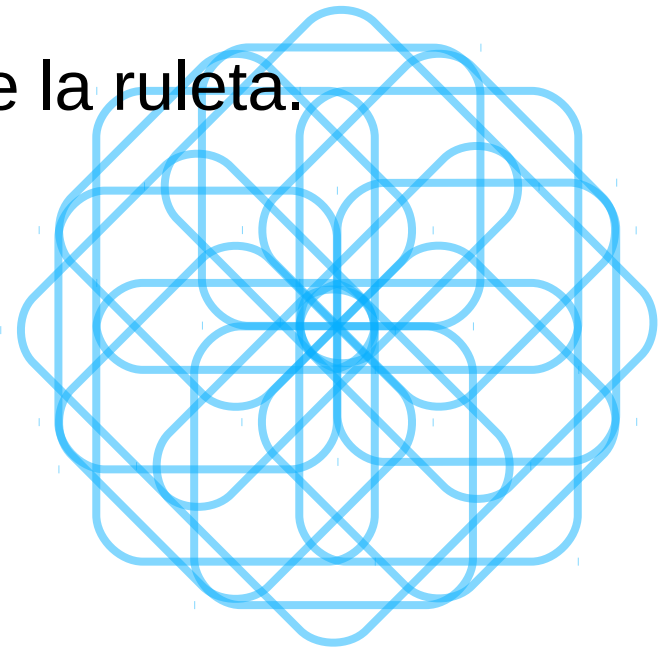
*1001*01



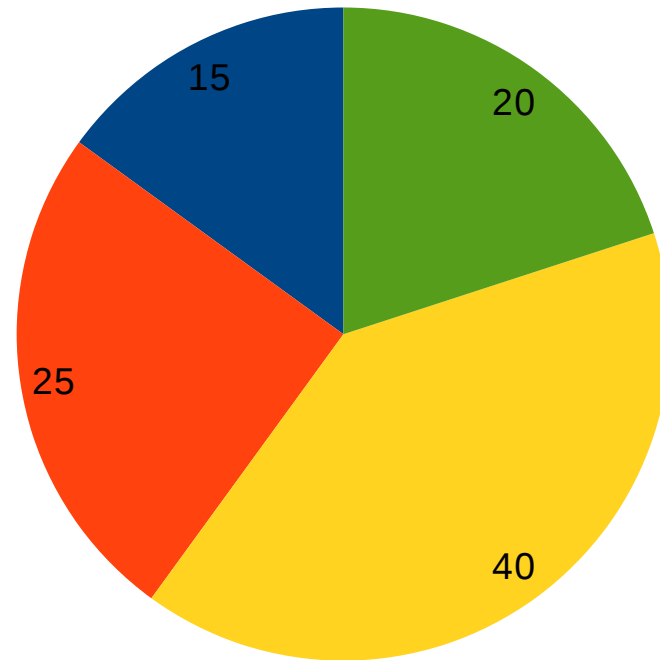
# Selección

La selección de qué individuos cruzar se lleva a cabo con base en la aptitud (fitness) de los individuos de la población. Los individuos más aptos son más propensos a reproducirse y generar crías.

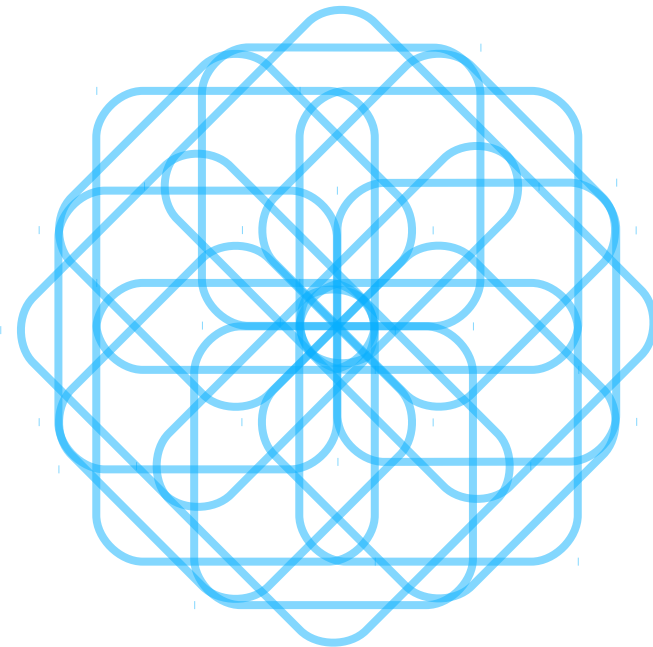
Un método común para la selección es el de la ruleta.



# Selección



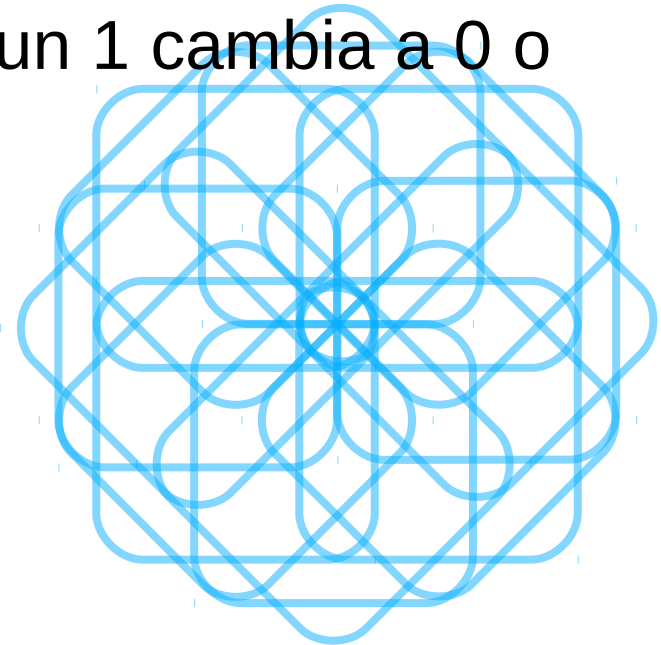
- Individuo 1
- Individuo 2
- Individuo 3
- Individuo 4



# Mutación

Es el último paso del algoritmo genético.

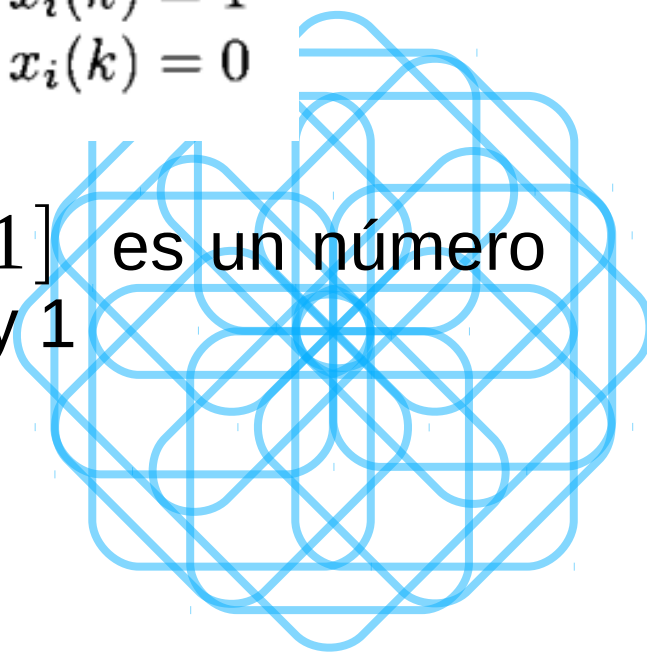
Para implementarla se selecciona una probabilidad de mutación, por ejemplo del 1%. Esto significa que después del proceso de cruce cada bit de cada cría tiene un 1% de probabilidad de cambiar al valor opuesto (un 1 cambia a 0 o un 0 cambia a 1)



Si se tiene una población de  $N$  individuos  $x_i$ , donde cada individuo tiene  $n$  bits y el factor de mutación es  $\rho$ , entonces al final de cada generación se invierte cada bit de cada individuo con una probabilidad de  $\rho$

$$r \leftarrow U[0, 1]$$
$$x_i(k) \leftarrow \begin{cases} x_i(k) & \text{if } r \geq \rho \\ 0 & \text{if } r < \rho \text{ and } x_i(k) = 1 \\ 1 & \text{if } r < \rho \text{ and } x_i(k) = 0 \end{cases}$$

Para  $i \in [1, N]$  y  $k \in [1, n]$  donde  $U[0, 1]$  es un número aleatorio uniformemente distribuido entre 0 y 1



# Algoritmo de Ruleta

$x_i$  = i-ésimo individuo en la población  $i \in [1, N]$

$f_i \leftarrow \text{fitness}(x_i)$  para  $i \in [1, N]$

$f_{\text{suma}} = \sum_{i=1}^N f_i$

generar número aleatorio  $r \in [0, f_{\text{suma}}]$

$F \leftarrow f_1$

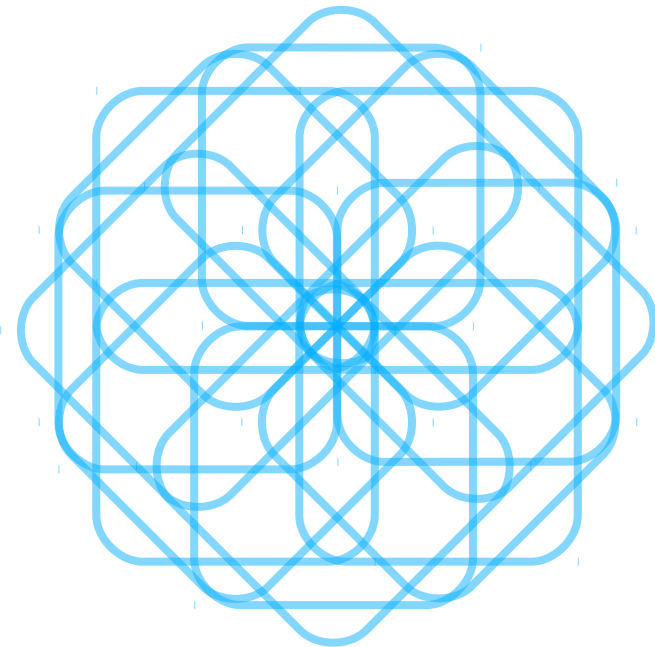
$k \leftarrow 1$

mientras  $F < r$

$k \leftarrow k + 1$

$F \leftarrow F + f_k$

$\text{padre} \leftarrow x_k$



# Algoritmo Genético

Padres  $\leftarrow$  {Población generada aleatoriamente}  
mientras no (criterio de paro)  
    calcular la aptitud de cada individuo en la población  
    Hijos  $\leftarrow \emptyset$   
    mientras  $|\text{Hijos}| < |\text{Padres}|$   
        usar aptitud para seleccionar un par de padres  
        cruzar a los padres para generar las crías c1 y c2  
        Hijos  $\leftarrow$  Hijos  $\cup$  {c1, c2}  
    mutar aleatoriamente algunos hijos  
Padres  $\leftarrow$  Hijos

