GRAY WOLF OPTIMIZATION.

Levels →

1. Alpha
2. Beta
3. Delta
4. Omega

~~Delta~~

1. Alpha → Leader
   → Responsible for decision making.
   → Acknowledged by others.

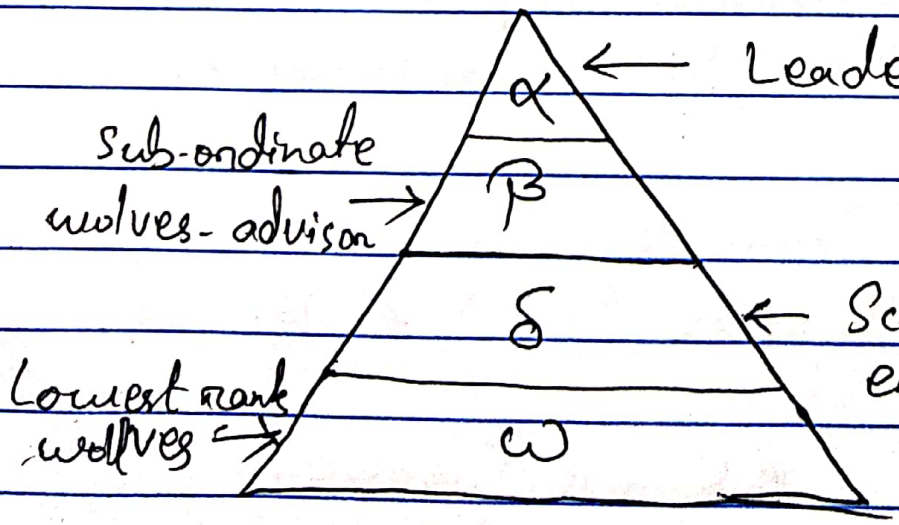
2. Beta → Help Alpha in decision making.
   → Advisor to Alpha.
   → Choosing Alpha
   → Discpline maintainance.

3. Delta → Dominate Omega
   → Reports to α & β.
   → Categorizes -
   * Scouts → Responsible for watching boundary.

* Sentinels → responsible for protecting the pack.
* Elders → Those who were α & β sometime.
* Hunters → Helps alpha & beta in hunting.
* Caretakers → Caring for ill weak & wounded wolves.

B. Omega → Are like scapegoat in the pack
→ Are last allowed wolves to eat.

Social Hierarchy :-



Search Process :-

1. The search process is modeled with the aim of mimicking the hunting behavior of gray wolf making use of three stages, searching, encircling and attacking the prey.
2. The first two stages are dedicated to exploration and the last one covers the exploitation.

Steps in GWO :-

1. Searching (exploration) → Finding the pray.
2. Encircle (exploration) → During hunting first then encircle pray.
3. Attacking (exploitation) → Usually guided by alpha, beta and delta might participate accordingly.

   * So in GWO first three best sol^{n} are saved and rest are oblige to update their position according to the position of best search agents.

1. Searching (Exploration) :-

* Grey wolves mostly search according to the position of the alpha, beta, and delta. They diverge from each other to search for prey and converge to attack prey.
* In GWO it is modeled by utilizing A with random values greater than 1 or less than -1 so the search agent could diverge from the pray.
* When $|A| > 1$ wolves are forced to diverge from the pray and find better one.

2. Encircling prey :-

* As mentioned above, grey wolves encircle prey during the hunt. In order to mathematically model encircling behavior the following equations are proposed -

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_p(t) - \vec{X}(t) \right|$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D}$$

where t indicates the current iteration, $\vec{A}$ & $\vec{C}$ are co-efficient vectors, $\vec{X}_p$ is the position vector of the prey, and $\vec{X}$ indicates the position vector of a grey wolf.

The vectors $\vec{A}$ & $\vec{C}$ are calculated as follows

$$\vec{A} = 2\vec{a}\vec{r}_1 - \vec{a}$$

$$\vec{C} = 2\vec{r}_2$$

$\vec{a}$ → Linearly decreased from 2 to 0.

(r_1, r_2) → Random vectors in $[0, 1]$.

3. Attacking :-

* The hunt is usually ~~do~~ guided by the alpha.
* The beta and delta might also participate in hunting occasionally.
*

$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}|$$

$$\vec{D}_\beta = |\vec{C}_2 \vec{X}_\beta - \vec{X}|$$

$$\vec{D}_\delta = |\vec{C}_3 \vec{X}_\delta - \vec{X}|$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 (\vec{D}_\alpha)$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2 (\vec{D}_\beta)$$

$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3 (\vec{D}_\delta)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3}$$

* Gray Wolf finish the hunt by attacking the pray when stop moving.
* In GWO vector A is a random value I interval [-2a, 2a] a decreased from 2 to over the course of iteration & when $|A| < 1$ the wolves attack the pray.

Algorithm :-

1. Initialize x_i = ~~Random numbers~~ (10, 12 etc)
2. Initialize a, A & C.

$$\vec{A} = 2\vec{a}(rand_1) - \vec{a}$$

$$\vec{C} = 2(rand_2)$$

$\vec{a}$ → Linearly decreased from 2 to 0 over the course of iterations.

3. Calculate fitness of each search agent

$$\vec{D} = \left| \vec{C}\ \vec{G}_p(t) - \vec{G}(t) \right|$$

$$\vec{G}(t+1) = \vec{G}_p(t) - \vec{A}\vec{D}$$

$$\vec{D}_\alpha = \left| \vec{C}_1 \cdot \vec{G}_\alpha - \vec{G} \right|$$

$$\vec{D}_\beta = \left| \vec{C}_2 \cdot \vec{G}_\beta - \vec{G} \right|$$

$$\vec{D}_\delta = \left| \vec{C}_3 \cdot \vec{G}_\delta - \vec{G} \right|$$

$$\vec{G}_1 = \vec{G}_\alpha - \vec{A}_1 \cdot (\vec{D}_\alpha)$$

$$\vec{G}_2 = \vec{G}_\beta - \vec{A}_2 \cdot (\vec{D}_\beta)$$

$$\vec{G}_3 = \vec{G}_\delta - \vec{A}_3 \cdot (\vec{D}_\delta)$$

X_α = best search agent.
X_β = second best "
X_δ = third best "

while (itr--) {
for (~~α, β, δ~~ i:pop) {
Update position of current search agent.

$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}|$$
$$\vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}|$$
$$\vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}|$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1(\vec{D}_\alpha)$$
$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2(\vec{D}_\beta)$$
$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3(\vec{D}_\delta)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3}$$

}
Update a, A & C
Calculate fitness of all agents.
Upate X_α, X_β & X_δ
~~t=t+1~~
}

return X_α.

$n = 10;$

$a = 2;$

$A = [2ar_{1,1} - a,\ 2ar_{2,1} - a,\ 2ar_{3,1} - a]$

$C = [2a_{12},\ 2a_{22},\ 2a_{32}]$

$x[1 \ldots 10] = rand().$

$f(x) = \sin(x)$

Best sol^n = X_α

2nd best = X_β

3rd best = X_δ

$D_\alpha = |C_1 X_\alpha - X(t)|$

$D_\beta = |C_2 X_\beta - X(t)|$

$D_\delta = |C_3 X_\delta - X(t)|$

$X_1 = X_\alpha - A_1(D_\alpha)$

$X_2 = X_\beta - A_2(D_\beta)$

$X_3 = X_\delta - A_3(D_\delta)$

$$x(t+1) = \frac{X_1 + X_2 + X_3}{3}$$