



**INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE CÓMPUTO**



**Algoritmos genéticos**

**“Selección por Ranking”**

**Alumno:**

**Navarro Pérez Rafael 3CM5**

## Índex

### Contenido

Introducción:.....	1
Desarrollo:.....	1
Resultados: .....	2
Conclusiones:.....	5

## Introducción:

Propuesta por Baker para evitar la convergencia prematura en las técnicas de selección proporcional. El objetivo de esta técnica es disminuir la presión de selección. En este caso, discutiremos el uso de jerarquías lineales, pero es posible también usar jerarquías no lineales, aunque la presión de selección sufre cambios más abruptos al usarse esta última.

Los individuos se clasifican con base en su aptitud, y se les selecciona con base en su rango (o jerarquía) y no con base en su aptitud. El uso de jerarquías hace que no se requiera escalar la aptitud, puesto que las diferencias entre las aptitudes absolutas se diluyen. Asimismo, las jerarquías previenen la convergencia prematura (de hecho, lo que hacen, es alentar la velocidad convergencia del algoritmo genético).

El algoritmo de las jerarquías lineales es el siguiente:

- Ordenar (o jerarquizar) la población con base en su aptitud, de **1** a **N** (donde 1 representa al menos apto).
- Elegir **Max** ( $1 \leq \text{Max} \leq 2$ )
- Calcular **Min** =  $2 - \text{Max}$
- El valor esperado de cada individuo será:

$$Valesp(i, t) = Min + (Max - Min) \frac{jerarquía(i, t) - 1}{N - 1}$$

Baker recomendó  $Max = 1.1$

Ejemplo: (jerarquía menor a la aptitud más baja)

	aptitud	jerarquías	$Valesp$	
(1)	12	2	0.95	
(2)	245	5	1.10	Aplicar ruleta u otra técnica proporcional
(3)	9	1	0.90	
(4)	194	4	1.05	
(5)	48	3	1.00	
			$\Sigma = 5.00$	

$$Max = 1.1 \quad Min = 2 - 1.1 = 0.9 \quad N = 5$$

$$Valesp = 0.9 + (0.2) \frac{jerarquía_i - 1}{N - 1}$$

## Desarrollo:

Para esta práctica se ocupó el IDE Code Blocks, y la herramienta Allegro.

esta practica consta con un total de 4 pruebas con diferentes cantidades de generaciones que son las siguientes 10, 30, 50, 100;

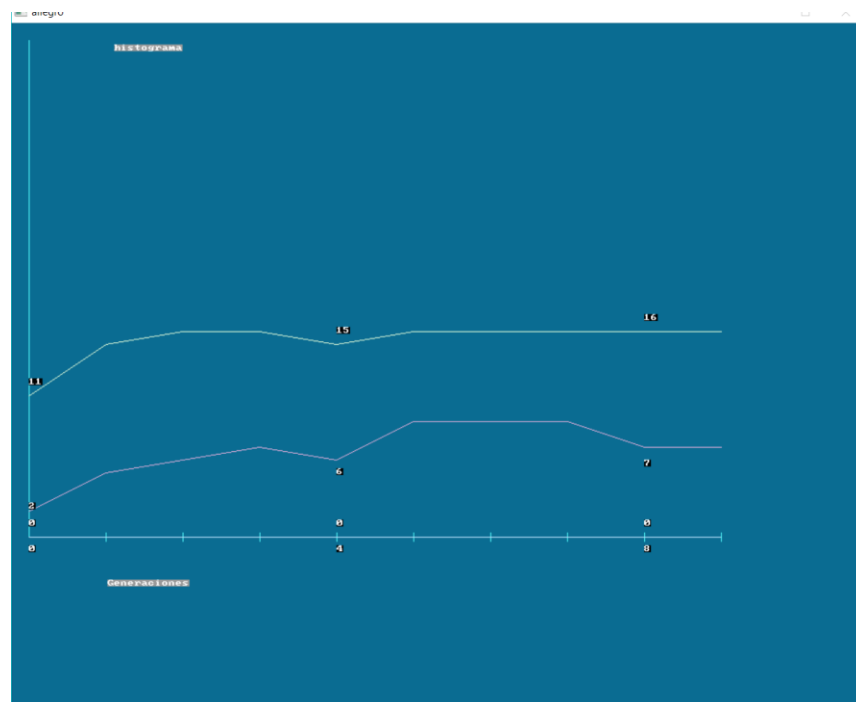
La siguiente función fue implementada para el objetivo de la función que es el ranking es decir acomodar los individuos de el menos apto al mas apto y posteriormente ocupar una forma de selección, que en su caso fue ruleta que se implanto anteriormente

```
1. void ranking(bitset<BIT_IND> individuo[],float res[]){
2.     float aux[NUM_IND];
3.     int n= sizeof(aux)/sizeof(float);
4.     for( int i =0; i<NUM_IND;i++){
5.         aux[i]= getIndividualAptitude(individuo[i]);
6.     }
7.     sort(aux,aux+n);
8.     float div,uno,dos,resta,multi;
9.     for( int i =0; i<NUM_IND;i++){
10.        uno=(i-1);
11.        dos=(NUM_IND-1);
12.        div = (uno/dos);
13.        resta= MAXIMO -(MINIM);
14.        multi= resta*div;
15.        res[i]=(float) MINIM + multi;
16.    }
17.    for( int i =0; i<NUM_IND;i++){
18.        printf("%f\n",res[i]);
19.        individuo[i]=res[i];
20.    }
21. }
```

### Resultados:

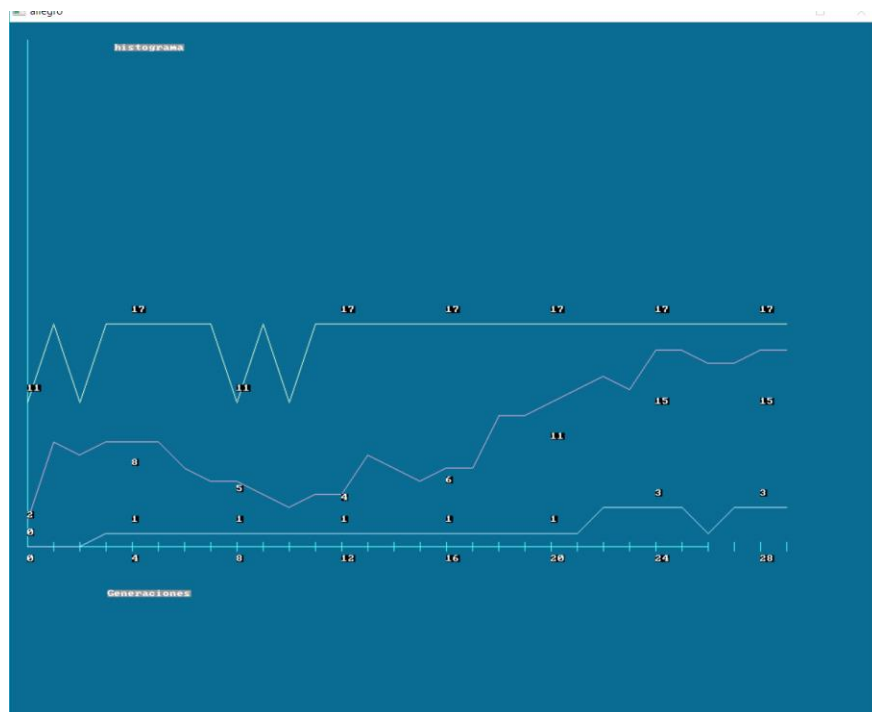
Los resultados mostrados son muy variados por el tipo de aptitud que se está ocupando así también por el tipo de selección de individuos que no se selecciona el más apto, Sino el que cumpla las condiciones con la ruleta.

La primera prueba fue hasta la generación número 10.



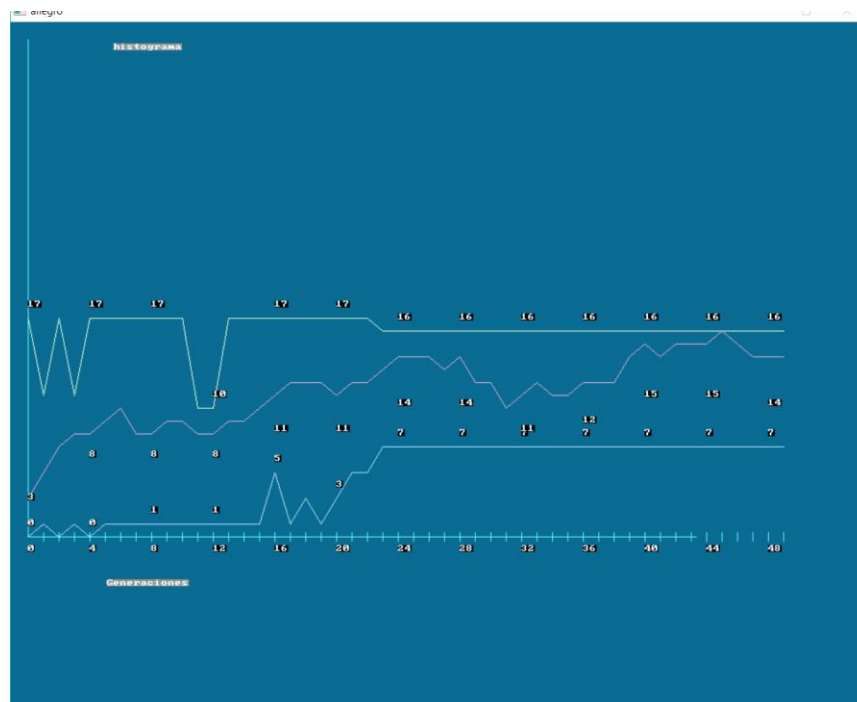
Como se logra observar hay una mejora significativa a comparación de la selección por ruleta simple ya que están acomodados de menor a mayor lo cual la probabilidad de seleccionar un individuo más apto sea más alta.

La siguiente prueba se realizó con hasta la generación 30:



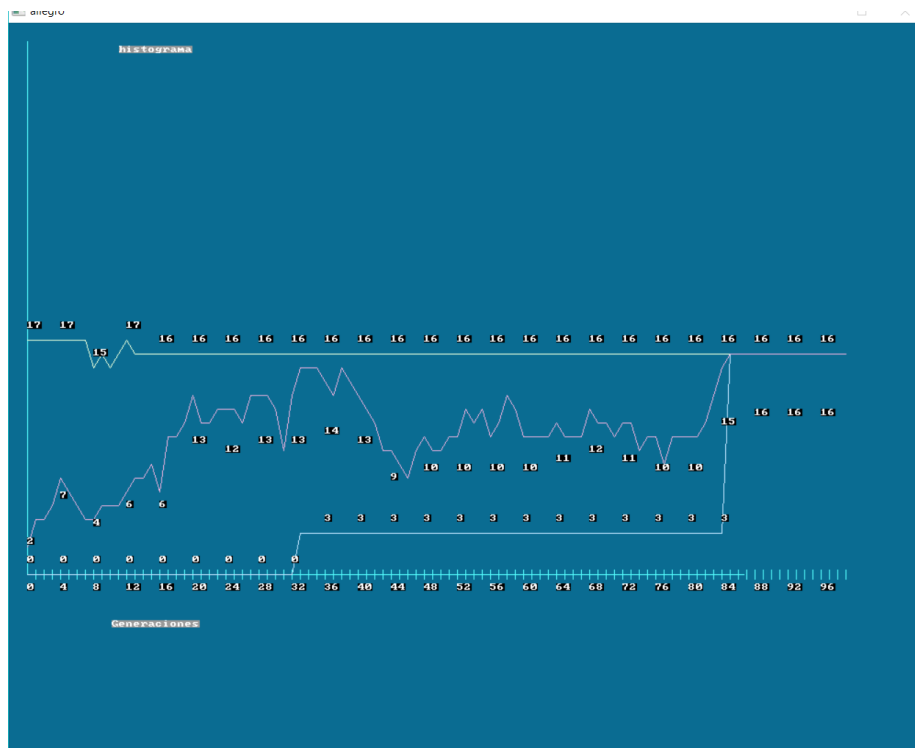
Se puede observar como la media va creciendo continuamente esto quiere decir que cada generación que pasa los individuos va mejorando.

La siguiente prueba se realizó hasta la generación 50:



En esta grafica se observa que el algoritmo va funcionando y no converge de manera prematura a comparación de los casos con ruleta y se va observando mejoras generación a generación.

Y la última prueba hasta la generación 100:



Se puede ver como después de N generaciones el algoritmo converge a un solo valor lo cual indica que ya se llegó al individuo más apto dada la función fitness de la aptitud

### **Conclusiones:**

En las tablas que salina como resultado se observaba en la selección desde un principio muchos valores seleccionados fueron sino los mas aptos unos de los mejores, como lo mencionaba anteriormente este método de selección no asegura escoger a los mas aptos desde un principio, pero si mejora pasando de generación en generación hasta llegar a un punto en donde la función tenga que converger y llegue al individuo mas apto para el caso.

También en esta practica pude observar que se puede crear un nuevo tipo de selección ya sea agregando pequeños detalles a un meto de selección como fue en el caso de esta práctica que se implementó una función antes de ejecutar la selección por ruleta dando así como resultado un mejor resultado que por si solo.