

<b>Periodo</b>	Enero-abril 2020
<b>Cuatrimestre y grupo</b>	8 A
<b>Asignatura</b>	Inteligencia artificial
<b>Corte</b>	1
<b>Actividad</b>	IA.C1.A1 Programación de Enfermeras Usando Algoritmo Genético
<b>Fecha de asignación</b>	21/01/2020
<b>Fecha de entrega</b>	31/01/2020
<b>Matriculas</b>	<b>Nombres</b>
<b>163211</b>	Ariel Eduardo Juárez Vázquez
<b>153224</b>	Alexis Pérez Gómez

#### **Actividad:**

Nurse Scheduling Using Genetic Algorithm: La programación de enfermeras en el hospital usa solo un formato que asigna la misma cantidad de enfermeras para atender a cada departamento cada semana. Este formato de programación no es consistente con el número de pacientes tratados cada día y no tiene en cuenta los factores relacionados con los niveles de servicio y los sistemas de colas en la programación de enfermeras. No es posible asignar tasas similares de pago de horas extras para todas las enfermeras. El investigador enfatiza la importancia de un modelo de programación de enfermeras apropiado y efectivo para garantizar el menor costo para el hospital, manteniendo el nivel de servicio del hospital y siendo justo al pagar horas extras a todas las enfermeras a tasas similares mediante el uso de un método de simulación y optimización para crear la programación de enfermeras más adecuada.

**Objetivo:** que Los gastos del hospital sean mínimos.

#### **Características:**

- la enfermera trabaja 60 horas como máximo por semana.
- las horas laborales como máximo son de 40 si trabaja más son horas extras.
- Horas máximos por día de 6 a 9 incluyendo horas extras.

El algoritmo tiene las siguientes funciones:

Números	FUNCIONES	Nombre descriptivo
1	<code>def poblacionInicial() :</code>	Población inicial
2	<code>def generandoHorasSalariadas() :</code>	Generando horas salariables
3	<code>def fitness() :</code>	Fitness
4	<code>def ruleta() :</code>	Ruleta
5	<code>def cruza_mutacion()</code>	Cruza y mutación
6	<code>def nuevosCandidatos() :</code>	Nuevos candidatos
7	<code>def graficar() :</code>	Graficar

Descripción de las funciones:

1. **Población inicial:** recibe como parámetro el número de enfermeras que tendrá cada individuo y el número de la población para iniciar, esta función se encarga de asignar a cada enfermera sus respectivas semanas de trabajo.
2. **Generando horas salariables :** recibe como parámetros la población y las semanas generadas anteriormente esta función es la encargada de asignarles departamento donde laborara la enfermera, horas de trabajo y tiempo extra, el departamento está representado con el numero 1..n, las horas de trabajo entre 6 horas y 9 diarias si trabaja tiempo extra será representado con un 1 en un vector de tres posiciones [1,8,0] siendo la posición 0 el departamento donde labora y la posición 1 las horas trabajadas la posición 2 si trabajo extra tendrá un 1 si no un 0, esto generaría los horarios de las horas trabajadas de las enfermeras.
3. **Fitness:** La función fitness recibe la población inicial, las semanas, los horarios de las horas trabajadas de las enfermeras, este método se encarga de obtener la cantidad total a pagar del tiempo normal y extra laborado de las enfermeras de cada individuo esto me obtiene una lista fitness que servirá en la ruleta.
4. **Ruleta:** la función ruleta recibe como parámetros la lista fitness, a cada uno de los individuos de la población se le asigna una parte proporcional a su fitness de una ruleta, de tal forma que la suma de todos los porcentajes sea la unidad. Los mejores individuos recibirán una porción de la ruleta mayor que la recibida por los peores. Generalmente la población está ordenada en base al fitness por lo que las porciones más grandes se encuentran al inicio de la ruleta. Para seleccionar un individuo basta con generar un número aleatorio del intervalo [0...100] y devolver el individuo situado en esa posición de la ruleta, este método presenta el inconveniente de que el peor individuo puede ser seleccionado más de una vez.
5. **Cruza y mutación:** Este método recibe dos padres quienes son elegidos por el método de la ruleta estos se cruzarán entre ellos seleccionando un punto a lazar en ambos padres cortando los padres en ese punto y creando hijos intercambiando las colas, este método implementado en el algoritmo se le conoce como (1-point crossover). La mutación tiene una probabilidad definida por el usuario el proceso de mutación consiste en generar un numero aleatorio si es menor o igual el porcentaje definido por el usuario muta en esta

caso el gen seria las horas trabajadas en tiempo normal de las enfermeras.

6. **Nuevos candidatos:** en esta función se generan los nuevos candidatos, la nueva población de enfermeras, semanas y horarios laborados que pasarían siendo la nueva generación para volver a repetir el ciclo hasta n generaciones.
7. **Graficar:** se encarga de graficar los mejores, peores y promedios del sueldo recibido por las enfermeras y así comprobando a través del algoritmo genético la optimización de los pagos de las enfermeras.

## Lenguaje utilizado y librerías:

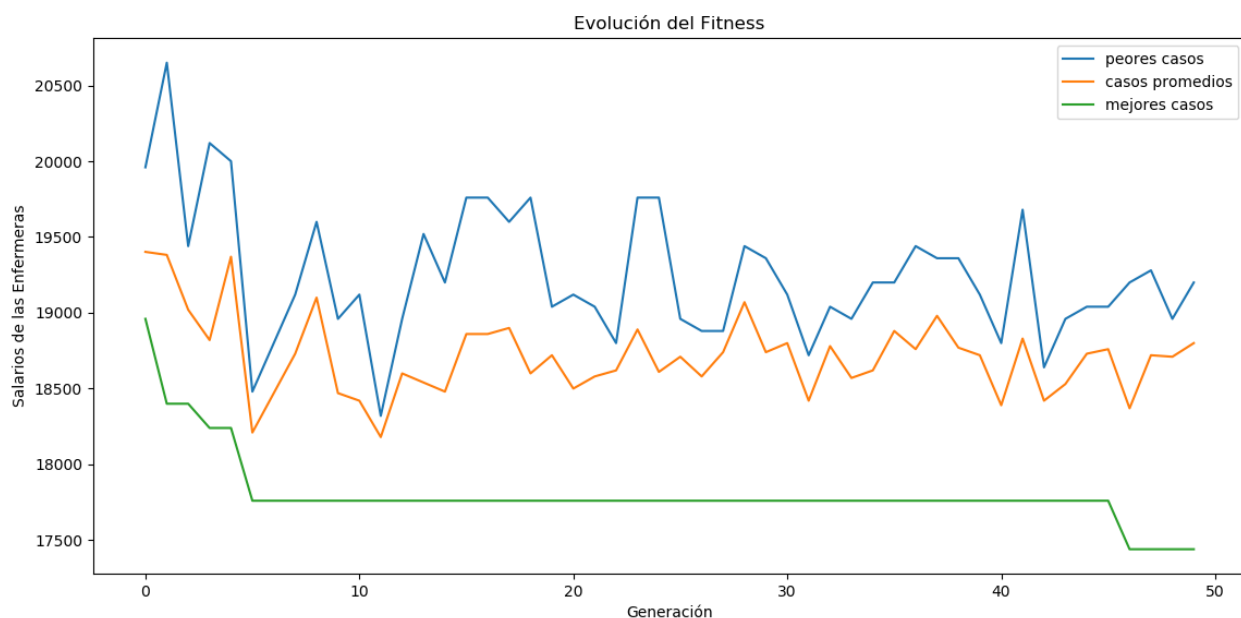
En la parte de programación se utiliza el lenguaje de programación Python.

**NumPy:** agrega mayor soporte para vectores y matrices, constituyendo una biblioteca de funciones matemáticas de alto nivel para operar con esos vectores o matrices.

**Matplotlib:** es una biblioteca para la generación de gráficos a partir de datos contenidos en listas o arrays en el lenguaje de programación Python y su extensión matemática NumPy.

**Random:** contiene una serie de funciones relacionadas con los valores aleatorios.

## RESULTADOS



En esta imagen se puede ver como la línea verde tiene un descenso en los gastos por cada generación a partir de la generación 5 0 6 el valor se

mantiene pero a partir de la 46 y valor vuelve a descender, en conclusión los gastos va 19,000\$ a 17,500\$

## Mejor candidato

[illegible]