



**INSTITUTO POLITÉCNICO
NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO**



ALGORITMOS GENÉTICOS

PRÁCTICA 8 OPERADORES DE MUTACIÓN PARA PERMUTACIONES

Profesora: Sandra Luz Morales Guitrón

Alumno: Alba Díaz Diego Samuel

Grupo: 3CM5

ÍNDICE DE CONTENIDO

Introducción	1
Contenido	1
Mutación por Inserción	1
Mutación por Desplazamiento	1
Mutación por Intercambio Recíproco	2
Mutación Heurística	2
Práctica 8.....	3
Descripción	3
Implementación de Mutación por Inserción	3
Compilación y Ejecución.....	3
Implementación de Mutación por Desplazamiento.....	4
Compilación y Ejecución.....	4
Implementación de Mutación por Intercambio Recíproco	5
Compilación y Ejecución.....	5
Implementación de Mutación Heurística	6
Compilación y Ejecución.....	6
Conclusión	7

Introducción

La mutación es la parte de un AG relacionada a la exploración del espacio de búsqueda. En términos simples, el operador de mutación puede ser definido como *un pequeño retoque al azar* en el cromosoma, para obtener una nueva solución. Es utilizado para mantener e introducir diversidad en la población y regularmente se utiliza con una baja probabilidad, si la probabilidad es muy alta el AG se reduce a una búsqueda al azar.

En algunos AG, los cromosomas se conforman por secuencias que no pueden tener dígitos repetidos. Esto hace a la mutación un poco más difícil, sin embargo, los experimentadores han creado todo tipo de métodos para mutar los cromosomas sin duplicar genes.

En esta práctica se estudiarán los siguientes operadores de mutación para la representación de permutaciones:

- Mutación por inserción.
- Mutación por desplazamiento.
- Mutación por intercambio recíproco.
- Mutación heurística.

Contenido

Mutación por Inserción

Se selecciona un gen al azar para ser desplazado e insertado nuevamente en el cromosoma, la posición a la cual se desplaza se elige también al azar. En las pruebas, este operador de mutación ha demostrado ser mejor que las otras alternativas mencionadas aquí.

Tomar a un gen al azar fuera de la secuencia:



Reinsertar al gen en una posición aleatoria:

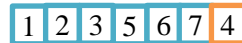


Ilustración 1. Ejemplo Mutación por Inserción.

Mutación por Desplazamiento

Es similar a la mutación por inserción, solo que se eligen una serie de genes a desplazar. Se seleccionan dos posiciones aleatorias y se toman los genes entre ellos como un grupo y, a continuación, se reinserta el grupo en una posición aleatoria.

Tomar un grupo al azar fuera de la secuencia:



Reinsertar el grupo de genes en una posición aleatoria:



Ilustración 2. Ejemplo de Mutación por Desplazamiento.

Mutación por Intercambio Recíproco

Se seleccionan dos posiciones al azar y se intercambian los genes en ellas.

Seleccionar dos genes al azar:

1	2	3	4	5	6	7
---	---	---	---	---	---	---

Intercambiar valores:

1	5	3	4	2	6	7
---	---	---	---	---	---	---

Ilustración 3. Ejemplo de Mutación por Intercambio Recíproco.

Mutación Heurística

Procedimiento:

1. Seleccionar λ genes al azar.
2. Generar posibles resultados de acuerdo a todas las permutaciones posibles de los genes seleccionados.
3. Evaluar los posibles resultados y seleccionar el más apto.

1)	P:	1	2	3	4	5	6	7	
	Permutaciones de:	2	4	5					
	a)	2	5	4					
	b)	4	2	5		c)	4	5	2
	d)	5	2	4		e)	5	4	2

2) Posibles resultados:

a)	1	2	3	5	4	6	7
b)	1	4	3	2	5	6	7
c)	1	4	3	5	2	6	7
d)	1	5	3	2	4	6	7
e)	1	5	3	4	2	6	7

3) Seleccionar a más apto. En este caso suponer que es d).

P:	1	5	3	2	4	6	7
----	---	---	---	---	---	---	---

Ilustración 4. Ejemplo de Mutación Heurística.

Práctica 8

Descripción

Programar los operadores de mutación descritos en esta práctica, sin importar el tamaño del cromosoma ni de la población.

Implementación de Mutación por Inserción

Compilación y Ejecución

En la *Figura 1* y *Figura 2* se anexan capturas de pantalla con la compilación y ejecución del código de programa desarrollado para el operador de mutación por inserción. La ejecución se realiza para una población con tamaño igual a 10 y la longitud del cromosoma es igual a 8.

En la *Figura 2* se resumen los resultados obtenidos en 3 columnas. En la primera columna se observa el individuo antes de mutar. La segunda columna tiene el valor a mover y la posición en de reinserción. Finalmente, en la tercera columna se aprecia el individuo mutado.

```
C:\> Símbolo del sistema
C:\Users\diego\Desktop\AG\practica8>gcc insertion.c nprmutation.c -o insertion
C:\Users\diego\Desktop\AG\practica8>_
```

Figura 1. Compilación – Mutación por Inserción.

```
C:\> Símbolo del sistema
C:\Users\diego\Desktop\AG\practica8>insertion 10 8
individuo: 1 5 8 7 2 4 6 3 | valor: 8, insertar en: 0 | mutacion: 8 1 5 7 2 4 6 3
individuo: 8 6 2 5 1 4 3 7 | valor: 5, insertar en: 7 | mutacion: 8 6 2 1 4 3 7 5
individuo: 7 8 2 1 4 6 5 3 | valor: 8, insertar en: 2 | mutacion: 7 2 8 1 4 6 5 3
individuo: 6 1 2 4 3 5 7 8 | valor: 4, insertar en: 5 | mutacion: 6 1 2 3 5 4 7 8
individuo: 8 6 7 2 5 4 3 1 | valor: 2, insertar en: 0 | mutacion: 2 8 6 7 5 4 3 1
individuo: 3 1 4 2 5 7 8 6 | valor: 6, insertar en: 6 | mutacion: 3 1 4 2 5 7 6 8
individuo: 8 3 6 5 1 7 4 2 | valor: 4, insertar en: 7 | mutacion: 8 3 6 5 1 7 2 4
individuo: 7 3 2 8 1 5 6 4 | valor: 2, insertar en: 6 | mutacion: 7 3 8 1 5 6 2 4
individuo: 3 4 1 7 6 2 8 5 | valor: 6, insertar en: 1 | mutacion: 3 6 4 1 7 2 8 5
individuo: 1 3 5 8 2 7 4 6 | valor: 7, insertar en: 2 | mutacion: 1 3 7 5 8 2 4 6
```

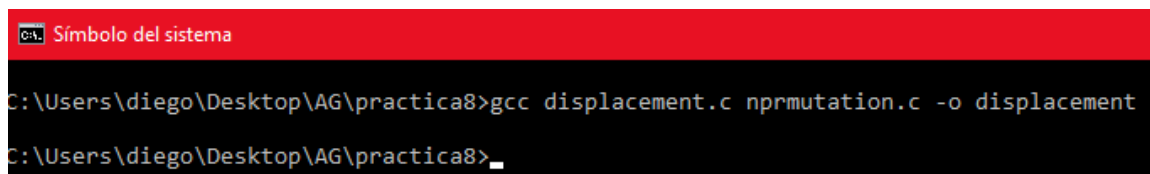
Figura 2. Ejecución – Mutación por Inserción.

Implementación de Mutación por Desplazamiento

Compilación y Ejecución

En la *Figura 3* y *Figura 4* se anexan capturas de pantalla con la compilación y ejecución del código de programa desarrollado para el operador de mutación por desplazamiento. La ejecución se realiza para una población con tamaño igual a 10 y la longitud del cromosoma es igual a 8.

En la *Figura 4* se resumen los resultados obtenidos en 3 columnas. La primera columna contiene al individuo a mutar. En la segunda columna se observan el grupo de genes a mover y la posición de reinserción. Y en la tercera columna se puede apreciar el individuo mutado.

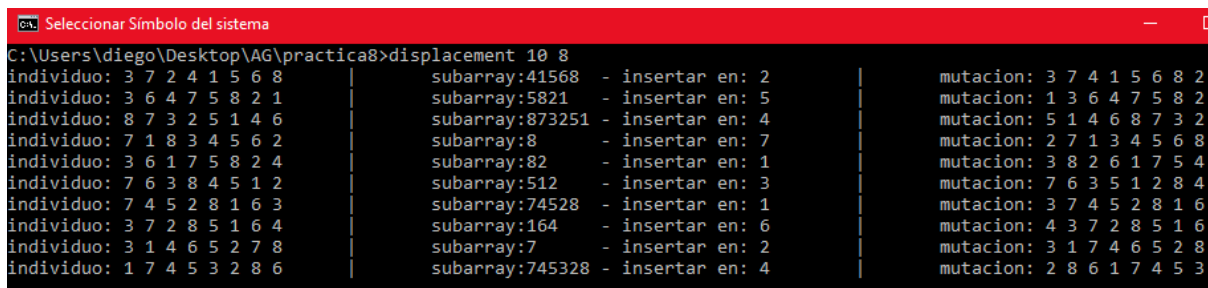


```

C:\Users\diego\Desktop\AG\practica8>gcc displacement.c nprmutation.c -o displacement
C:\Users\diego\Desktop\AG\practica8>_

```

Figura 3. Compilación – Mutación por Desplazamiento.



```

C:\Users\diego\Desktop\AG\practica8>displacement 10 8

```

individuo: 3 7 2 4 1 5 6 8	subarray:41568 - insertar en: 2	mutacion: 3 7 4 1 5 6 8 2
individuo: 3 6 4 7 5 8 2 1	subarray:5821 - insertar en: 5	mutacion: 1 3 6 4 7 5 8 2
individuo: 8 7 3 2 5 1 4 6	subarray:873251 - insertar en: 4	mutacion: 5 1 4 6 8 7 3 2
individuo: 7 1 8 3 4 5 6 2	subarray:8 - insertar en: 7	mutacion: 2 7 1 3 4 5 6 8
individuo: 3 6 1 7 5 8 2 4	subarray:82 - insertar en: 1	mutacion: 3 8 2 6 1 7 5 4
individuo: 7 6 3 8 4 5 1 2	subarray:512 - insertar en: 3	mutacion: 7 6 3 5 1 2 8 4
individuo: 7 4 5 2 8 1 6 3	subarray:74528 - insertar en: 1	mutacion: 3 7 4 5 2 8 1 6
individuo: 3 7 2 8 5 1 6 4	subarray:164 - insertar en: 6	mutacion: 4 3 7 2 8 5 1 6
individuo: 3 1 4 6 5 2 7 8	subarray:7 - insertar en: 2	mutacion: 3 1 7 4 6 5 2 8
individuo: 1 7 4 5 3 2 8 6	subarray:745328 - insertar en: 4	mutacion: 2 8 6 1 7 4 5 3

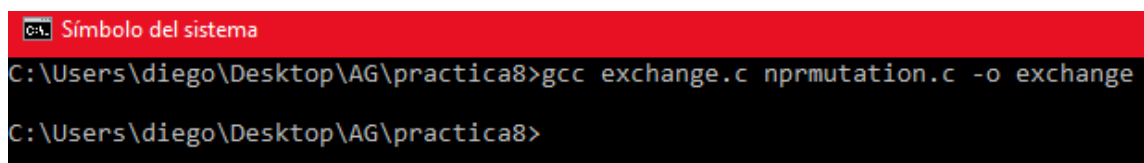
Figura 4. Ejecución - Mutación por Desplazamiento.

Implementación de Mutación por Intercambio Recíproco

Compilación y Ejecución

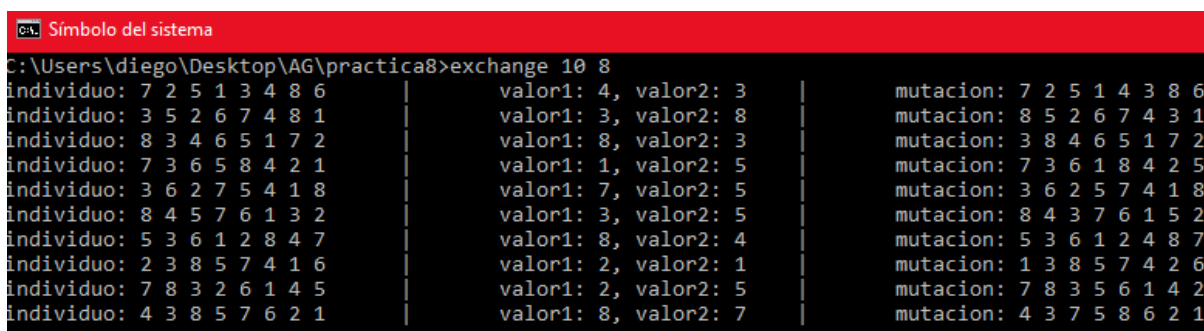
En la *Figura 5* y *Figura 6* se anexan capturas de pantalla con la compilación y ejecución del código de programa desarrollado para el operador de mutación por intercambio recíproco. La ejecución se realiza para una población con tamaño igual a 10 y la longitud del cromosoma es igual a 8.

En la *Figura 6* se resumen los resultados obtenidos en 3 columnas. La primera columna contiene al individuo a mutar. En la segunda columna se notan los genes a intercambiar. Y en la tercera columna se puede apreciar el individuo mutado.



```
C:\Users\diego\Desktop\AG\practica8>gcc exchange.c nprmutation.c -o exchange  
C:\Users\diego\Desktop\AG\practica8>
```

Figura 5. Compilación – Mutación por Intercambio Recíproco.



```
C:\Users\diego\Desktop\AG\practica8>exchange 10 8  
individuo: 7 2 5 1 3 4 8 6 | valor1: 4, valor2: 3 | mutacion: 7 2 5 1 4 3 8 6  
individuo: 3 5 2 6 7 4 8 1 | valor1: 3, valor2: 8 | mutacion: 8 5 2 6 7 4 3 1  
individuo: 8 3 4 6 5 1 7 2 | valor1: 8, valor2: 3 | mutacion: 3 8 4 6 5 1 7 2  
individuo: 7 3 6 5 8 4 2 1 | valor1: 1, valor2: 5 | mutacion: 7 3 6 1 8 4 2 5  
individuo: 3 6 2 7 5 4 1 8 | valor1: 7, valor2: 5 | mutacion: 3 6 2 5 7 4 1 8  
individuo: 8 4 5 7 6 1 3 2 | valor1: 3, valor2: 5 | mutacion: 8 4 3 7 6 1 5 2  
individuo: 5 3 6 1 2 8 4 7 | valor1: 8, valor2: 4 | mutacion: 5 3 6 1 2 4 8 7  
individuo: 2 3 8 5 7 4 1 6 | valor1: 2, valor2: 1 | mutacion: 1 3 8 5 7 4 2 6  
individuo: 7 8 3 2 6 1 4 5 | valor1: 2, valor2: 5 | mutacion: 7 8 3 5 6 1 4 2  
individuo: 4 3 8 5 7 6 2 1 | valor1: 8, valor2: 7 | mutacion: 4 3 7 5 8 6 2 1
```

Figura 6. Ejecución - Mutación por Intercambio Recíproco.

Implementación de Mutación Heurística

Compilación y Ejecución

En la *Figura 7* y *Figura 8* se anexan capturas de pantalla con la compilación y ejecución del código de programa desarrollado para el operador de mutación heurística. La ejecución se realiza para una población con tamaño igual a 10 y la longitud del cromosoma es igual a 8.

En la *Figura 8* se resumen los resultados obtenidos en 3 columnas. En la primera columna se observa el individuo antes de mutar. La segunda columna contiene los genes a permutar. Y finalmente, en la tercera columna se aprecia el individuo mutado.

```
C:\Users\diego\Desktop\AG\practica8>gcc heuristic.c nprmutation.c -o heuristic
C:\Users\diego\Desktop\AG\practica8>
```

Figura 7. Compilación – Mutación Heurística.

```
C:\Users\diego\Desktop\AG\practica8>heuristic 10 8
individuo: 1 4 7 6 3 8 2 5 | genes: 876 | mutacion: 1 4 6 8 3 7 2 5
individuo: 3 4 1 5 2 6 8 7 | genes: 26835 | mutacion: 3 4 1 2 5 6 8 7
individuo: 7 6 4 2 5 3 1 8 | genes: 2753 | mutacion: 5 6 4 3 2 7 1 8
individuo: 7 1 6 8 4 5 2 3 | genes: 26 | mutacion: 7 1 2 8 4 5 6 3
individuo: 6 8 3 4 2 7 5 1 | genes: 7456 | mutacion: 4 8 3 7 2 5 6 1
individuo: 7 4 8 3 1 2 6 5 | genes: 651 | mutacion: 7 4 8 3 5 2 1 6
individuo: 5 7 2 3 6 8 4 1 | genes: 743 | mutacion: 5 4 2 3 6 8 7 1
individuo: 4 5 2 6 1 7 3 8 | genes: 42 | mutacion: 2 5 4 6 1 7 3 8
individuo: 6 7 4 1 8 2 5 3 | genes: 253176 | mutacion: 3 5 4 1 8 7 6 2
individuo: 7 3 1 6 4 2 8 5 | genes: 4685217 | mutacion: 6 3 8 7 4 2 1 5
```

Figura 8. Ejecución – Mutación Heurística.

Conclusión

Mediante esta práctica se desarrollaron cuatro operadores de mutación distintos para representación por permutaciones. Como se ha observado los métodos de mutación desarrollados están especialmente diseñados para mutar cromosomas sin duplicar genes. Así, al igual que en la *Práctica No. 7*, se resalta la importancia de seleccionar un método de mutación de acuerdo a las características del problema a resolver y el tipo de representación de cromosoma elegido. Igualmente se reafirma la función del operador de mutación como la parte de un AG que permite ampliar el espacio de búsqueda, manteniendo la diversidad en la población.