MemorIa Práctica 2

Autores:

Acosta Novas Yhondri Josué

Ignacio Nicolás López

**Implementación**

Para la implementación de esta práctica hemos tomado el esqueleto de la práctica 1 como base.

Como cambio destacable tenemos que la representación de los individuos ahora es una lista de enteros.

Este cambio lo hemos llevado a cabo para facilitarnos la tarea a la hora de hacer los cruces y mutaciones de la población.

**Algoritmos de cruce implementados**

* Cruce por ciclos (CX)
* Cruce por codificación ordinal
* Cruce por orden OX
* Cruce por recombinación de rutas
* PMX

**Algoritmos de mutación implementados**

* Mutación heurística
* Mutación por inserción
* Mutación por intercambio
* Mutación por inversión

**Algoritmos de selección implementados (práctica 1)**

* Ranking
* Ruleta
* Torneo
* Truncamiento
* Estocástico
* Restos

**Problema Datos12**

**A screenshot of a cell phone

Description automatically generated**

Imagen – Gráfico que ilustra el resultado óptima alcanzado y los parámetros necesarios

**Tipo de cruce – PMX**

Consiste en elegir un tramo de uno de los progenitores y cruzar preservando el orden y la posición de la mayor cantidad de ciudades del otro.

A screenshot of a computer

Description automatically generated

Imagen – Implementación del cruce PMX.

**Mutación por inversión**

Este tipo de mutación se aplica con una determinada probabilidad y consiste en seleccionar dos puntos el individuo al azar e invertir los elementos que hay entre dichos puntos.

A screenshot of a cell phone

Description automatically generated

Imagen – Implementación mutación por inversión.

La siguiente tabla muestra los resultados de ejecutar 20 veces el problema Datos12 con los mismos parámetros.

A screenshot of a cell phone

Description automatically generated

Destacar que podríamos haber disminuido el número de generaciones ya que como podemos apreciar en el gráfico de la solución, esta se consigue alrededor de la generación 550 – 600.

**Problema Datos15**

A screenshot of a computer

Description automatically generated

Imagen – Gráfico que ilustra el resultado óptima alcanzado y los parámetros necesarios

**Cruce por codificación ordinal**

Codificamos los genes de los individuos y después los cruzamos aplicando el cruce clásico. Por último descodificamos los genes.

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

Imágenes cruce por codificación ordinal.

La siguiente tabla muestra los resultados de ejecutar 20 veces el problema Datos15 con los mismos parámetros.

A close up of a piece of paper

Description automatically generated

En esta ocasión hemos tenido que aumentar considerablemente el tamaño de la población para poder conseguir el resultado buscado. Por último hemos ajustado correctamente el número de generaciones, ya que el resultado se consigue alrededor de la generación 450 y el número de generaciones total era de 500.

**Problema Datos30**

A screenshot of a computer

Description automatically generated

Imagen – Ilustra el ejercicio Datos30 donde hemos conseguido una aproximación del resultado.

Cruce por ciclos

1. Tomamos la primera posición nueva en el padre P1.
2. Buscamos el en la misma posición de P2.
3. Vamos a la posición con el mismo gen en P1.
4. Añadimos este gen al ciclo.
5. Repetimos los pasos 2-4 hasta que completemos el nuevo individuo.

**A screenshot of a cell phone

Description automatically generated**

Imágenes cruce por ciclos.

La siguiente tabla muestra los resultados de ejecutar 20 veces el problema Datos30 con los mismos parámetros.

**A close up of a piece of paper

Description automatically generated**

**Destacable:** La curva de cambio a lo largo de las generaciones es bastante estable la diferencia entre el mejor y el peor valor a lo largo de las generaciones cada vez se hace más pequeña.

El mejor valor se consigue en una generación temprana, alrededor de la 350, por lo que aumentar el número de generaciones no ayudaría a conseguir una mejor aproximación del problema.

También probamos con conjunto de parámetros (algoritmo de selección, cruce, mutación diferentes), pero solo conseguíamos peores resultados de media que se alejaban muchísimo del resultado objetivo.

**Otros algoritmos implementados:**

**Cruce YI:**

Este algoritmo de cruce, crea un nuevo individuo cruzando los genes de forma aleatoria de ambos padres.

**Mutación YI:**

Esta mutación intercambia n posiciones elegidas al azar de los genes del padre.

Ajuntamos documento Excel que contiene las tablas con los datos de las ejecuciones de los 3 problemas.