Algoritmos genéticos Ejercicio de Serpientes:

**Masa**

Máximo 4000 gr  
Mínimo 500 gr

Se ocupan de 1 a 12 dígitos binarios, ocuparemos los 12

**Velocidad**

Máximo 8.3 m/s  
Minimo 0 m/s

Se ocupan de 1 a 10 dígitos binarios, pero ocupamos 12  
Entonces un string de 24 dígitos expresa ambas características.

Elegimos usar la máxima cantidad de dígitos que necesita una característica para representar a la otra también extendiendo también la cantidad de dígitos que necesita (Con 0s a la izquierda) para que así podamos identificar fácilmente ambas características de forma fácil. Sabiendo que cada característica está representada por 12 dígitos. (En este caso son 12, porque para expresar el número más grande, 4000 se ocupaban 12).

**Mi serpiente**

|  |  |  |
| --- | --- | --- |
| **Característica** | **Característica decimal** | **Característica Binaria** |
| **Masa** | **520gr** | 001000001000 |
| **Velocidad** | **2.45** | 0010 00101101 |

**La unión de ambas característica es:**  
001000001000 000010101101

La representación de un número con fracción (decimales) en binario se hace de una forma muy sofisticada, pero para nuestro caso, lo hemos hecho más simple, simplemente dedicándole cierta parte de los dígitos a la parte entera, y el resto a la parte decimal.  
En este caso, la mayor cantidad de dígitos que íbamos a ocupar para representar el máximo número decimal (8) eran 4 dígitos binarios. Y la mayor cantidad de dígitos que íbamos a necesitar parar representar el máximo número decimal (99) eran 8 dígitos binarios.

Así que la parte entera está formada de 4 dígitos, y la parte decimal está hecha con 8 dígitos: 00010 00101101

**Variación de masa 10 números diversos:**

**Variación de velocidad 10 números diversos:**

0 a 8.3 en pasos de 0.92 debería ser:

* Numero1: 0
* Numero2: 0.92222222
* Numero3: 1.84444444
* Numero4: 2.76666666
* Numero5: 3.68888888
* Numero6: 4.6111111
* Numero7: 5.53333332
* Numero8: 6.45555554
* Numero9: 7.37777776
* Numero10: 8.299

Pero nosotros escribimos:

* Numero1: 0 0000 0000000
* Numero2: 0.92 0000 1011100
* Numero3: 1.84 0001 1010100
* Numero4: 2.76 0010 1001100
* Numero5: 3.68 0011 1000100
* Numero6: 4.61 0100 0111101
* Numero7: 5.53 0101 0110101
* Numero8: 6.45 0110 0101101
* Numero9: 7.37 0111 0100101
* Numero10: 8.3 1000 0000011

**Ejercicio de Algoritmos genéticos.**Dada una población:

1. 11110010
2. 10101011
3. 00111000
4. 10000111
5. 01001110
6. 00101111
7. 00101100
8. 01101111

La función de evaluación va a ser por cuartetos. Cada individuo tiene 2 cuartetos. Y se evalúa cada cuarteto en función de ciertas reglas:

* Por cada 1 que haya: Se suma 1
* Por cada “par de 1s” consecutivos Se suma 1
* Si el cuarteto tiene más 1s que 0s Se suma 1
* Si el cuarteto tiene más 0s que 1s Se resta 1
* Por cada 0 que haya en el cuarteto: Se resta 1
* Por cada par de 0s consecutivos: Se resta 1

Individuo B: Primer Cuarteto: 1010  
Cantidad de 1s: +2  
Cantidad de 0s: -2  
Cantidad 1s consecutivos: 0  
Cantidad de 0s Consecutivos: 0  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? 0  
SubTotal: 0

Individuo B: Segundo Cuarteto: 1011  
Cantidad de 1s: +3  
Cantidad de 0s: -1  
Cantidad 1s consecutivos: +1  
Cantidad de 0s Consecutivos: 0  
¿Más 1s que 0s?: +1  
¿Más 0s que 1s? 0  
SubTotal: 4

Total: 4

Individuo A: Primer Cuarteto: 1111  
Cantidad de 1s: +4  
Cantidad de 0s: 0  
Cantidad 1s consecutivos: +3  
Cantidad de 0s Consecutivos: 0  
¿Más 1s que 0s?: +1  
SubTotal: 4 +3 +1 = 8

Individuo A: Segundo Cuarteto: 0010  
Cantidad de 1s: +1  
Cantidad de 0s: -3  
Cantidad 1s consecutivos: 0  
Cantidad de 0s Consecutivos: -1  
¿Más 1s que 0s?: -1  
SubTotal: 1 – 3 -1 -1 = -4

Calificación total de A: 4

Individuo C: 00111000

Individuo C: Primer cuarteto 0011  
Cantidad de 1s: +2  
Cantidad de 0s: -2  
Cantidad 1s consecutivos: +1  
Cantidad de 0s Consecutivos: -1  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? 0  
SubTotal: 0

Individuo C: Segundo Cuarteto: 1000  
Cantidad de 1s: +1  
Cantidad de 0s: -3  
Cantidad 1s consecutivos: 0  
Cantidad de 0s Consecutivos: -2  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? -1  
SubTotal: -5

Total: -5

Individuo D: 10000111

Individuo D: Primer cuarteto 1000  
Cantidad de 1s: 1  
Cantidad de 0s: -3  
Cantidad 1s consecutivos: 0  
Cantidad de 0s Consecutivos: -2  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? -1  
SubTotal: -5

Individuo D: segundo cuarteto 0111  
Cantidad de 1s: +3  
Cantidad de 0s: -1  
Cantidad 1s consecutivos: +2  
Cantidad de 0s Consecutivos: 0  
¿Más 1s que 0s?: +1  
¿Más 0s que 1s? 0  
SubTotal: 5

Total 0

Individuo E 01001110

Individuo E: Primer cuarteto 0100  
Cantidad de 1s: +1  
Cantidad de 0s: -3  
Cantidad 1s consecutivos: 0  
Cantidad de 0s Consecutivos: -1  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? -1  
SubTotal: -4

Individuo E: segundo cuarteto 1110  
Cantidad de 1s: +3  
Cantidad de 0s: -1  
Cantidad 1s consecutivos: +2  
Cantidad de 0s Consecutivos: 0  
¿Más 1s que 0s?: +1  
¿Más 0s que 1s? 0  
SubTotal: 5

Total: 1

Individupo F: 00101111

Individuo F: segundo cuarteto 0010  
Cantidad de 1s: +1  
Cantidad de 0s: -3  
Cantidad 1s consecutivos: 0  
Cantidad de 0s Consecutivos: -1  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? -1  
SubTotal: -4

Individuo F: segundo cuarteto 1111  
Cantidad de 1s: +4  
Cantidad de 0s: 0  
Cantidad 1s consecutivos: +3  
Cantidad de 0s Consecutivos: 0  
¿Más 1s que 0s?: +1  
¿Más 0s que 1s? 0  
SubTotal: 8

Total: 4

Individuo H 01101111  
Individuo H: segundo cuarteto 0110  
Cantidad de 1s: +2  
Cantidad de 0s: -2  
Cantidad 1s consecutivos: +1  
Cantidad de 0s Consecutivos: 0  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? 0  
SubTotal: 1

Individuo H: segundo cuarteto 1111  
Cantidad de 1s: +4  
Cantidad de 0s: 0  
Cantidad 1s consecutivos: +3  
Cantidad de 0s Consecutivos: 0  
¿Más 1s que 0s?: +1  
¿Más 0s que 1s? 0  
SubTotal: 8

Total: 9

Individuo G: 00101100

Individuo G: segundo cuarteto 0010  
Cantidad de 1s: +1  
Cantidad de 0s: -3  
Cantidad 1s consecutivos: 0  
Cantidad de 0s Consecutivos: -1  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? -1  
SubTotal: -4

Individuo G: segundo cuarteto 1100  
Cantidad de 1s: +2  
Cantidad de 0s: -2  
Cantidad 1s consecutivos: +1  
Cantidad de 0s Consecutivos: -1  
¿Más 1s que 0s?: 0  
¿Más 0s que 1s? 0  
SubTotal: 0

Total: -4

Producto Integrador de Aprendizaje

Al igual que en el ejercicio, nuestros pasos a seguir para crear el algoritmo (esta vez en R) que va a operar el Algoritmo genético los pasos son:

1. **Generación de la Población Inicial**:
   * Crear una función que genere una población inicial de individuos, donde cada individuo es una cadena binaria de 8 caracteres.
2. **Función de Evaluación**:
   * Implementar una función que separe cada cadena en dos mitades de 4 caracteres.
   * Evaluar cada mitad según las reglas dadas:
     + Si hay un **1** en la cadena: sumar 1.
     + Si hay dos **1**s seguidos en la cadena: sumar 1.
     + Si hay más **1**s que **0**s en la cadena: sumar 1.
     + Si hay un **0** en la cadena: restar 1.
     + Si hay dos **0**s seguidos en la cadena: restar 1.
     + Si hay más **1**s que **0**s en la cadena: restar 1.
   * Calcular la puntuación total del individuo sumando las puntuaciones de ambas mitades.
3. **Selección de los Mejores Individuos**:
   * Ordenar los individuos en función de sus puntuaciones.
   * Seleccionar los dos individuos con las mejores puntuaciones.
4. **Cruce de Individuos**:
   * Cruzar los dos mejores individuos seleccionados, combinando sus mitades de las siguientes formas:
     + Primera mitad del primer individuo con la segunda mitad del segundo individuo.
     + Segunda mitad del segundo individuo con la primera mitad del primer individuo.
   * Generar nuevos individuos a partir de estas combinaciones y elegir el más apto de cada cruce.
5. **Generación de la Nueva Población**:
   * Continuar cruzando los más aptos hasta obtener una nueva población de la misma cantidad de individuos con la que se inició.
   * No es necesario que todos se crucen, sólo los más aptos. De hecho, no se van a cruzar todos. En nuestro caso del ejercicio hecho previamente, para el tamaño de la población inicial, no se cruzaba uno.
6. **Mutación**:
   * Implementar una función de mutación que invierta los valores de la diagonal de individuos en la nueva población.

Partes del algoritmo:

Primeramente, el paso 1 era generar una población inicial. En este caso, una población de 7 individuos con combinaciones aleatorias de 0s y 1s, que tengan en total 8 caracteres.

En mi caso hice primero una función para