

Manual de usuario

Usted eligió leer este manual para saber como jugar al juego de la vida diseñado por el grupo InvalidRead. Para comenzar el juego usted debe ingresar en un archivo .txt cada una de las células que estarán vivas al comenzar el juego, poniendo en la primera linea el tamaño del tablero, en la segunda linea la célula con su posición en x y en y y todos sus genes (pueden ser mas de uno). Las debe ingresar en el .txt una abajo de la otra de la siguiente forma:

```
tablero 300 100
```

```
celula 20 30
```

```
gen 1000100010001111000001 20
```

```
gen 10001001010101011110010001 80
```

```
gen 1001111 90
```

```
fin
```

```
celula 21 30
```

```
gen 1000000111 70
```

```
gen 1001111 20
```

```
fin
```

```
celula 22 30
```

```
gen 1000100010001111000001 30
```

```
gen 10001001010101011110010001 90
```

```
gen 1001111 10
```

```
fin
```

El tablero estará indicado en la primera linea, con posición en X y en Y. A partir de aquí, usted solo debe comenzar los turnos, donde en cada turno se mostrara el tablero y una serie de datos pedidos.

Puede llamarle la atención que tenga que ingresar un dato llamado genes, pero no debe darle mayor importancia, solo debe escribir una combinación de unos y ceros, y separado por un espacio su “intensidad”. Las células pueden estar vivas o muertas, y eso va a estar determinado en el tablero. **Al comenzar el juego y luego de ejecutar cada turno, se creara una imagen bitmap con una ilustración del tablero.** Usted puede seleccionar la opción ejecutar turno, que le mostrara ciertos datos como cantidad de células vivas, promedio de células muertas, etc. Puede reiniciar la partida, o puede dar por finalizado el juego. También tiene una opción para comenzar o terminar el seguimiento de un gen, esta opción le mostrara como varia la carga genética de un gen desde que comienza a seguirlo hasta que decide dejar de seguirlo. La ilustración será en un grafico cartesiano en imagen bitmap.

INFORME DEL TP

Decidimos que el objeto Tablero sea un array bidimensional dinámico con cada uno de sus valores que sea un objeto Célula, ya que nos serviría de mucha utilidad para poder enlazar los objetos entre ellos (el objeto Gen y informacionGenetica también).

Para contabilizar y poder retener los datos de las células vecinas(para utilizarlos en las operaciones de transferencia y mutación) utilizamos un sinónimo de cambio de estado. Por ejemplo: si una célula estaba viva y pasa a muerta, el sinónimo es CasiMuerta, entonces así sabemos que estaba viva y pasa a muerta, para no perder el dato de que anteriormente al turno estaba viva.

Utilizamos una lista simplemente enlazada porque para la utilidad que nosotros le íbamos a dar (guardar los genes,intensidades y una lista de 3 células vecinas), utilizar una doble enlazada iba a acomplejar un poco el código y no tenia sentido.

Supusimos que la intensidad acumulada era la suma de todas las intensidades.

IMPORTANTE:

A pesar de que sabemos que queda memoria sin liberar, no estamos encontrando el motivo por el que sucede. Están todos los destructores, y también liberamos todos los bloques necesarios.Como aclaracion, les avisamos que liberarIntensidades en Gen.h y liberarGenes en Celula.h, hacen que la aplicación pierda menos memoria(no entendemos el motivo) pero tira errores por double free. Algo que no encontramos significado, ya que en realidad estos métodos hacen lo mismo que hace el destructor de lista.

MANUAL DEL PROGRAMADOR

La aplicación esta resueltos en 14 clases, que son:

- Tablero
- Célula
- Gen
- Información genética
- Interfaz usuario
- Juego de la vida
- Lista
- Informes
- Intensidad
- Color
- Bits
- Impresora de imágenes
- Seguimiento
- Gen seguimiento

Juego de la vida

En la clase juego de la vida se va a procesar los archivos y inicializar el juego. Esta clase inicializa el tablero y comienza los turnos, muestra los informes, la interfaz de la consola, etc.

Tablero

En tablero se van a calcular las células vecinas vivas, se va a actualizar la condición de las células y va a crear un triple puntero a célula que tendrá la matriz bidimensional, que luego apunta a un

puntero a célula. Su tamaño puede ser variable dependiendo de lo escrito en el archivo .txt inicial.

Lista

Es una lista simplemente enlazada.

INFORMES

En esta clase, se tendrán todos los contadores requeridos para calcular los datos pedidos al final de cada turno, de atributos privados. También estarán sus métodos para reiniciar contadores, sumar valores, mostrar variables.

INTERFAZ USUARIO

Esta clase simplemente imprime mensajes en consola para el usuario.

CELULA

Aquí van a estar todos los atributos de las células, sus genes, su estado, y muchos de sus métodos serán condiciones necesarias del TP, como calcularIntensidad, o completarTransferencia, por ejemplo.

GEN

En gen se van a acumular todos los datos del gen(información genética, intensidad, lista de intensidades, edad del gen). No realiza métodos importantes, pero tiene muchísima información.

INFORMACIÓN GENÉTICA

En esta clase se van a tener los dato como la cadena(bits)y el tamaño de la cadena que son los bits. En el mismo se van a hacer los procedimientos de combinación de bits.

INTENSIDAD

Acá van a estar los atributos de la intensidad, como su carga genetica.

BITS

Este objeto, básicamente inicializa una imagen con determinado tamaño, asigna un color a determinado pixel, y escribe el archivo.bmp

COLOR

En color vamos a poder asignar colores a determinados objetos según condiciones, por ejemplo para una célula, asignaremos verde o rojo según este viva o muerta, así podemos distinguirlas luego en el tablero que dibujamos en el bmp.

IMPRESORA DE IMAGENES

Acá estará todo lo necesario para poder imprimir las imágenes bmp, recorriendo posición a posición en el tablero.

GEN SEGUIMIENTO

Este gen sera el elegido por el usuario para hacerle un seguimiento, y guardara la carga acumulada del gen.

SEGUIMIENTO

Aca se van a tener todos los datos del gen a seguir, como el turno donde fue pedido el seguimiento, donde se termino, la informacion genetica del gen, etc.