

75.41 - 95.15 - Algoritmos y Programación II

Cátedra Ing. Patricia Calvo - 1er cuatrimestre 2018

Trabajo Práctico 2: Granjeros V1.0

Grupo NEO - Integrantes:

Joel Pérez

Gabriela Choque

Verónica Leguizamón

Analía Ludueña

Noelia Rodriguez

Desarrollamos una aplicación en C++ que implementa el juego de Consola “Granjeros”.

En base a lo indicado por el enunciado, se realizaron las siguientes suposiciones:

- * Cada vez que se solicita al jugador seleccionar una parcela, se ingresa primero fila y después columna.
- * El agua disponible en cada turno esta almacenada en un tanque cuya capacidad es determinada por el tamaño del terreno y el nivel de dificultad, en caso de quedar acumulada agua del anterior turno y en el turno actual haber recibido agua de mas, dicho sobrante se pierde (el tanque queda con su tope máximo de unidades).
- * Las parcelas se representan en pantalla de la siguiente forma:

Vacía: _____#

Cultivo Sembrado: _____(nombre del cultivo)

Parcela en Recuperación: _____R

Parcela con Cultivo Seco: _____S

Parcela con Cultivo Podrido: _____P

Cuestionario:

1. ¿Que es un Debug?

Debug (depuración) es el proceso de encontrar y resolver errores en un programa.

2. ¿Que es un “Breakpoint”?

Un Breakpoint (punto de interrupción) es una marca que ponemos en una o mas lineas de código para que cuando estamos haciendo debug y la ejecución del programa pase justamente por esa linea, se detenga y aguarde alguna instrucción nuestra para continuar.

3. ¿Que es “Step Into”, “Step Over”, “Step Out”?

- Step Into (Entrar): Ejecuta la siguiente linea de código, en caso de haber una llamada a un procedimiento, ingresa a dicho procedimiento y ejecuta la siguiente linea dentro de este.
- Step Over (Pasar por encima): Ejecuta la siguiente linea, en caso de haber una llamada a un procedimiento, lo ejecuta sin ingresar en este, como si solo fuera una instrucción mas.
- Step Out (Salir): Ejecuta el resto de lineas de código que quedan por ejecutar.

Manual del Usuario:

Pueden jugar una o más personas.

Una vez ejecutado el programa se ingresan la cantidad de jugadores, los nombres de los jugadores, las dimensiones del Terreno(NxM) y la dificultad del juego.

[Ejemplo visual] Al iniciar el juego se muestra en pantalla lo siguiente, por ejemplo:

```
Turno Nro.: 1
Jugador: pepe
Credito Disponible: 50
Estado del Terreno:
# # # # #
# # # # #
# # # # #
# # # # #
# # # # #
Numero del Dado: 5
Unidades de Riego recibidas: 25
Agua contenida en el Tanque: 25
Ingrese una Acción a realizar:
1 - Sembrar semillas en una parcela.
2 - Cosechar una parcela.
3 - Regar una parcela.
5 - Envió.
6 - Comprar un Terreno.
7 - Vender un Terreno.
8 - Comprar capacidad del tanque de agua.
9 - Comprar capacidad del almacén.
10- Avanzar al siguiente turno.
```

Cada vez que se muestra el terreno en pantalla (de forma textual) también se genera un archivo .bmp en el directorio screenshots (dentro del directorio del ejecutable). Este se abre con la aplicación predeterminada del Sistema Operativo, para poder visualizarla en ese mismo momento. Nota: para mostrar la imagen utiliza el comando xdg-open, en caso de no tenerlo instalado, es necesario ejecutar la siguiente instrucción en una ventana de terminal: `sudo apt install xdg-utils`

El terreno esta representado por una grilla de N filas por M columnas, las cuales representan las parcelas de dicho terreno. Se usan los siguientes símbolos según el estado de cada parcela:

Vacia: _____#
Cultivo Sembrado:_____ (nombre del cultivo)
Parcela en Recuperación:_____R
Parcela con Cultivo Seco:_____S
Parcela con Cultivo Podrido:_____P

Al iniciar cada turno se tira (automáticamente) un dado, y en base al numero que salga, este se multiplica por 5 y eso da la cantidad de unidades de riego para cargar en el tanque (que a su vez dependen de la dificultad seleccionada).

El tanque de agua almacena hasta un máximo de unidades, superado dicho limite el agua remanente se pierde y no se puede utilizar.

En este momento el jugador puede realizar cuantas acciones desee sobre cualquier parcela, hasta que decida saltar de turno, o bien abandonar el juego:

1 - Sembrar semillas en un parcela:

La primer acción que puede realizar un jugador es sembrar una cosecha a su elección, la cual le va a costar una determinada cantidad de créditos. Solo se puede sembrar una parcela cuando esta esta vacía.

2 - Regar una parcela:

Cada vez que se realiza una Siembra, o se arranca un nuevo turno, es necesario regar todas y cada una de las parcelas cultivadas, caso contrario, las que no fueron regadas al siguiente turno se secan y se pierden, teniendo que esperar un turno mas para volver a cultivar. Solo se puede regar una misma parcela una vez cada turno.

3 - Cosechar una parcela:

Luego de llegar al tiempo de maduración de un cultivo, es necesario cosecharlo, dado que si no se realiza eso, dicho cultivo se pudre y se pierde, aunque su tiempo de recuperación para volver a sembrar se reduce a la mitad. Dicha cosecha se envía a un almacén para ser enviada a su destino futuramente.

4 - Enviar una cosecha a un destino:

Cuando una cosecha se encuentra en el almacen, puede ser enviada a un destino, que es el momento en el cual se recibe la ganancia por dicha cosecha, y como contrapartida tiene un costo determinado.

5 - Comprar un terreno:

En caso de tener el credito suficiente, el jugador puede comprar un nuevo terreno para poder seguir sembrando mas cultivos. Este terreno tiene un costo de compra que va aumentando a medida que el jugador va adquiriendo nuevos terrenos.

6 - Vender un terreno:

El jugador puede vender cada uno de sus terrenos para obtener una ganancia, aunque el valor de venta es del 50% del valor original de compra.

7 - Comprar mas capacidad para el tanque:

Cuando el o los tanques que tiene un jugador ya no son suficientes para regar el o los terrenos que posee, puede comprar un nuevo tanque para almacenar el agua que recibe en cada turno. De esta manera el agua remanente que perdía cuando su tanque se llenaba, ahora puede agregarla en un nuevo tanque.

8 - Comprar mas capacidad para el almacén:

Esta opción es útil cuando un jugador tiene mas cultivos para cosechar que capacidad en su almacen actual.

9 - Cambiar de Terreno:

Dado que cada jugador puede administrar uno o mas terrenos, esta opción permite seleccionar con cual de esos terrenos va a trabajar en este momento.

Luego de haber realizado todas las acciones que considere conveniente el jugador, debe seleccionar la opción 10, para poder avanzar al siguiente turno (o la opción 11 para abandonar definitivamente el juego, en tal caso los restantes jugadores pueden continuar). Así sucesivamente hasta que se llega al ultimo turno (el cual fue seleccionado al iniciar el juego).

Manual del Programador:

El proyecto consta de los siguientes archivos:

- **Carpeta lib:**
Allí se encuentra la librería EasyBMP, la cual se utiliza para generar imágenes en formato BMP del terreno de juego, por cada jugador, por cada turno.
- **Lista.h y Nodo.h:**
Contienen la implementación de la Lista Simplemente Enlazada.
- **TP2.cpp y TP2.h:**
Contiene el método main el cual ejecuta el TDA Juego.
- **Constantes.h:**
Contiene las constantes que determinan las configuraciones y valores utilizados a lo largo del programa.
- **EstadoParcela.h:**
Enumerado que contiene los estados que puede tener una parcela a lo largo de su vida útil.
- **TipoDificultad.h:**
Enumerado con los 3 niveles de dificultad del juego.
- **Almacen.cpp y Almacen.h:**
TDA encargado de almacenar cosechas.
- **Archivo.cpp y Archivo.h:**
Se encarga de las operaciones con archivos, rutas, crear carpetas, etc.
- **Consola.cpp y Consola.h:**
Se encarga de mostrar por pantalla información y solicitarla por teclado.
- **Cultivo.cpp y Cultivo.h:**
TDA que maneja la información de cultivos.
- **Destino.cpp y Destino.h:**
TDA que maneja la información de destinos.
- **Dificultad.cpp y Dificultad.h:**
Contiene la información necesaria para configurar la dificultad del juego.
- **Imagen.cpp y Imagen.h:**
Realiza las operaciones para la generación de la imagen del terreno de cada turno de cada jugador.
- **Juego.cpp y Juego.h:**
Contiene todas las funciones principales del juego y su bucle principal de ejecución de turnos y jugadores.
- **Jugador.cpp y Jugador.h:**
TDA que maneja la información del jugador.
- **Parcela.cpp y Parcela.h:**
Métodos para realizar las acciones sobre cada parcela.
- **Tanque.cpp y Tanque.h:**
TDA que maneja el agua de cada jugador.
- **Terreno.cpp y Terreno.h:**
TDA que contiene una lista de parcela que simboliza el terreno.
- **Texto.cpp y Texto.h:**
Funciones útiles para administrar strings.

Nota: En las propiedades del proyecto, es necesario configurar el siguiente post build step:

```
cp -r ../res . ; cp -r ../datos . ;
```

Esto es para que las carpetas que contienen las imágenes y los archivos de datos sean copiadas a la ruta de ejecución del programa principal.