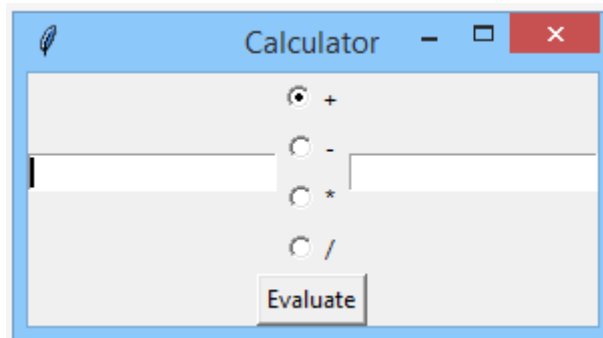


## Ejercicio 1

Necesitas una calculadora. Una calculadora muy sencilla y muy específica. Mire la imagen: contiene dos campos que el usuario puede usar para ingresar argumentos, un botón de radio para seleccionar la operación a realizar y un botón que inicia la evaluación:



Esperamos que la calculadora se comporte de la siguiente manera:

- Si ambos campos contienen números válidos (enteros o flotantes), hacer clic en el botón Evaluar debería mostrar una ventana de información que muestra el resultado de la evaluación;
- Si alguno de los campos contiene datos no válidos (p. ej., una cadena o un campo está vacío), al hacer clic en el botón Evaluar debería aparecer una ventana de error que describa el problema y el enfoque debería moverse al campo que causa el problema.

No olvide proteger su código para que no se divida por cero y use el administrador de cuadrícula para componer el interior de la ventana.

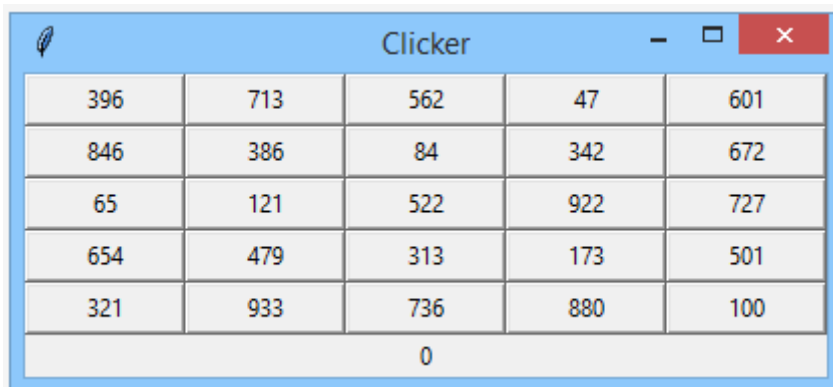
## Ejercicio 2

Queremos que escribas un juego simple pero desafiante, que puede ayudar a muchas personas a mejorar sus habilidades de percepción y memoria visual. Llamaremos al juego The Clicker ya que hacer clic es lo que esperamos del jugador.

El tablero de Clicker consta de 25 botones y cada uno de los botones contiene un número aleatorio del rango 1..999. Nota: ¡cada número es diferente!

Debajo del tablero hay un temporizador que inicialmente muestra 0. El temporizador comienza cuando el usuario hace clic en el tablero por primera vez.

Así es como imaginamos el estado inicial del tablero de The Clicker:



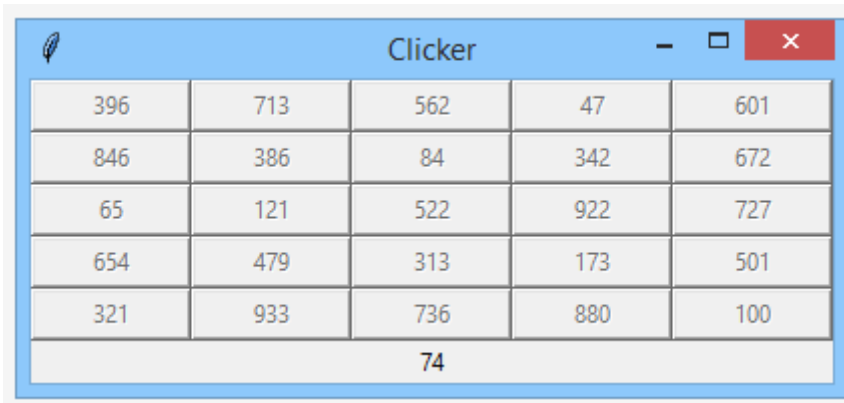
|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| 396 | 713 | 562 | 47  | 601 |
| 846 | 386 | 84  | 342 | 672 |
| 65  | 121 | 522 | 922 | 727 |
| 654 | 479 | 313 | 173 | 501 |
| 321 | 933 | 736 | 880 | 100 |
| 0   |     |     |     |     |

Esperamos que el jugador haga clic en todos los botones en el orden impuesto por los números, desde el más bajo hasta el más alto. Las reglas adicionales dicen que:

- El botón en el que se hizo clic correctamente cambia el estado del botón a DESHABILITADO (el botón se vuelve gris)
- El botón en el que se hizo clic incorrectamente no muestra actividad,
- El temporizador aumenta su valor cada segundo,

- Cuando todos los botones están atenuados (es decir, el jugador ha completado su tarea), el temporizador se detiene inmediatamente.

Así se ve el tablero cuando termina el juego:



The screenshot shows a Java Swing window titled "Clicker" with a standard Mac OS X title bar (blue with a pencil icon, and red, yellow, and green buttons). The window contains a 5x5 grid of numbers. The numbers in the grid are:

|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| 396 | 713 | 562 | 47  | 601 |
| 846 | 386 | 84  | 342 | 672 |
| 65  | 121 | 522 | 922 | 727 |
| 654 | 479 | 313 | 173 | 501 |
| 321 | 933 | 736 | 880 | 100 |

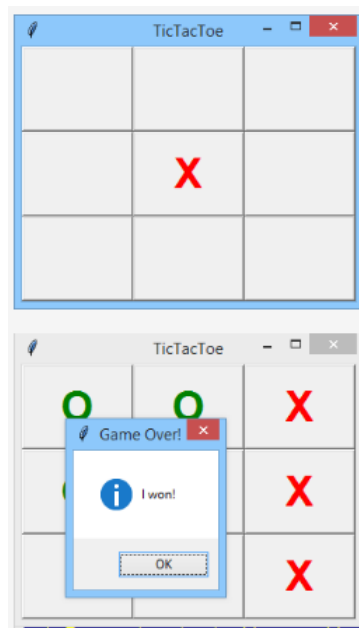
Below the grid, there is a single cell containing the number 74.

Sugerencia: considere usar el evento `<Button-1>` en lugar de configurar la propiedad del botón de comando; puede simplificar su código.

### Ejercicio 3

Escriba un programa GUI simple que pretenda jugar tres en raya con el usuario. No tengas miedo, no queremos que implementes algoritmos de inteligencia artificial. Puede hacerlo, si lo desea, pero preferimos concentrarnos en los problemas de la interfaz de usuario. Si realmente quieres crear un competidor real, hazlo por tu cuenta.

Así es como se ve el juego que está a punto de escribir (el comienzo y el final de muestra del juego):



Para hacer su tarea un poco más fácil, simplifiquemos un poco el juego. Aquí están nuestras suposiciones:.

- La computadora (es decir, su programa) reproduce 'X', y las X siempre son rojas,
- El usuario (por ejemplo, usted) juega 'O', y O siempre son verdes,
- El tablero consta de 9 fichas, y el papel de ficha se juega con un botón,
- El primer movimiento pertenece a la computadora - siempre pone su primera 'X' en el medio del tablero,
- El usuario ingresa su movimiento haciendo clic en la ficha elegida (hacer clic en las fichas que no están libres no es efectivo)
- El programa verifica si se cumplen las condiciones de fin del juego, y si el juego ha terminado, se muestra un cuadro de mensaje anunciando al ganador,
- de lo contrario, la computadora responde con su jugada y se repite la verificación,
- Use random para generar los movimientos de la computadora.