Figures Multithread

This is a simple package for creating and manipulating geometric figures. The user can add his own figures in accordance with the "Figure" interface. Figure manipulation is done in batch form using the "Batch" class. Batch runs commands that implement the "Command" interface. Batch can use sequential or multithread policy. The user can add his own command in accordance with the "Command" interface. Batch implements the "Execute" and "Validate" methods for executing commands and checking for overlapping figures after manipulations.

CanvasAtomicImpl

- m_x_size:int
- m_y_size:int
- m_canvas:vector<vector<atomwrapper<int>>>
- Resize(int x_size, int y_size)
- GetColor(int x, int y):int
- SetColor(int x, int y, T color)
- GetSize(int &x_size, int &y_size)
- + DrawCanvas()
- + std::unique ptr<CanvasImpl<T>> Clone()

CanvasNonAtomicImpl

- m_x_size:int
- m_y_size:int
- m_canvas:vector<vector<int>>
- + Resize(int x_size, int y_size)
- + GetColor(int x, int y):int
- + SetColor(int x, int y, T color)
- GetSize(int &x_size, int &y_size)
- + DrawCanvas()
- + std::unique ptr<CanvasImpl<T>> Clone()

Rectangle

- m_ulx:int
- m uly:int
- m_lrx:int
- 111_11 \.1111
- m_lry:int - m_color:int
- + IsNotOverlaped():bool
- + Clear()
- + Draw(int offset_x, int offset_y)
- + ReDraw(int color)
- + Clone():unique_ptr<Rectangle<int>>
- RectangleDraw(int color, int offset_x, int offset_y)

Circle

- m_x:int
- m_y:int
- m_radius:int
- m color:int
- + IsNotOverlaped():bool
- + Clear()
- + Draw(int offset_x, int offset_y)
- + ReDraw(int color)
- + Clone():unique_ptr<Circle<int>>
- CircleDraw(int color, int offset_x, int offset_y)

<<interface>> CanvasImpl

- + Resize(int x size, int y size)
- + GetColor(int x, int y):int
- + SetColor(int x, int y, T color) + GetSize(int &x_size, int &y_size)
- + DrawCanvas()
- + std::unique_ptr<CanvasImpl<T>> Clone()

