Target- Противник (Enemy)

Променливи потребни за исцртување на противниците:

public static int RadiusHead { get; set; } = 28;

public static int BodyWidth { get; set; } = 60;

public static int BodyHeight { get; set; } = 120;

public static int ArmWidth { get; set; } = 20;

public static int ArmHeight { get; set; } = 60;

public static int LegWidth { get; set; } = 20;

public static int LegHeght { get; set; } = 80;

public Point HeadCenter { get; set; }

public Point BodyCenter { get; set; }

public Point ArmLeft { get; set; }

public Point ArmRight { get; set; }

public Point LegLeft { get; set; }

public Point LegRight { get; set; }

public Point Crosshair { get; set; }

public static int CrosshairRadius { get; set; } = 7;

public Target()

{

}

Дефинирање на позициите на Points:

public void UpdatePoints()

{

BodyCenter = new Point(HeadCenter.X - RadiusHead, HeadCenter.Y + (BodyHeight / 2) - RadiusHead + 5);

ArmLeft = new Point(BodyCenter.X - BodyWidth / 2 + ArmWidth / 2, BodyCenter.Y);

ArmRight = new Point(BodyCenter.X + BodyWidth, BodyCenter.Y);

LegLeft = new Point(BodyCenter.X, BodyCenter.Y + BodyWidth \* 2);

LegRight = new Point(BodyCenter.X + BodyWidth - LegWidth, BodyCenter.Y + BodyWidth \* 2);

}

Цртање на противниците:

public void Draw(Graphics g)

{

Brush brush = new SolidBrush(Color.Cyan);

Pen pen = new Pen(Color.DarkCyan);

//Head

g.FillEllipse(brush, HeadCenter.X - RadiusHead, HeadCenter.Y - RadiusHead, RadiusHead \* 2, RadiusHead \* 2);

g.DrawEllipse(pen, HeadCenter.X - RadiusHead, HeadCenter.Y - RadiusHead, RadiusHead \* 2, RadiusHead \* 2);

//Body

brush = new SolidBrush(Color.DarkCyan);

g.FillRectangle(brush, BodyCenter.X, BodyCenter.Y, BodyWidth, BodyHeight);

//arms

brush = new SolidBrush(Color.DarkTurquoise);

g.FillRectangle(brush, ArmLeft.X, ArmLeft.Y, ArmWidth, ArmHeight);

g.FillRectangle(brush, ArmRight.X, ArmRight.Y, ArmWidth, ArmHeight);

//legs

brush = new SolidBrush(Color.DarkTurquoise);

g.FillRectangle(brush, LegLeft.X, LegLeft.Y, LegWidth, LegHeght);

g.FillRectangle(brush, LegRight.X, LegRight.Y, LegWidth, LegHeght);

//crosshair

pen = new Pen(Scene.account.CrossHairColor, Scene.account.CrossHairThickness);

g.DrawLine(pen, Scene.Pointer, new Point(Scene.Pointer.X, Scene.Pointer.Y + CrosshairRadius));

g.DrawLine(pen, Scene.Pointer, new Point(Scene.Pointer.X - CrosshairRadius, Scene.Pointer.Y));

g.DrawLine(pen, Scene.Pointer, new Point(Scene.Pointer.X + CrosshairRadius, Scene.Pointer.Y));

g.DrawLine(pen, Scene.Pointer, new Point(Scene.Pointer.X, Scene.Pointer.Y - CrosshairRadius));

if (Scene.account.CrossHairHaveCircle)

{

g.DrawEllipse(pen, Scene.Pointer.X - CrosshairRadius, Scene.Pointer.Y - CrosshairRadius, CrosshairRadius \* 2, CrosshairRadius \* 2);

}

pen.Dispose();

brush.Dispose();

}

Методи за проверка дали има погодок на одреден дел од телото на противникот

public bool HitHead(Point cursur)

{

int distance = (int)Math.Sqrt(Math.Pow(HeadCenter.X - cursur.X, 2) + Math.Pow(HeadCenter.Y - cursur.Y, 2));

return distance <= RadiusHead;

}

public bool HitBody(Point cursur)

{

if (cursur.X >= BodyCenter.X && cursur.X <= BodyCenter.X + BodyWidth && cursur.Y >= BodyCenter.Y && cursur.Y <= BodyCenter.Y + BodyHeight)

{

return true;

}

return false;

}

Методите HitLeftArm() HitRightArm() HitLeftLeg() HitRightLeg(): се многу слични