Карпенко О. КН20-1

**ЛАБОРАТОРНА РОБОТА 7**

Удосконалення програм емулятора дисплейного модуля і клієнта (спрайт).

Етапи виконання лабораторної роботи:

1. Розширити специфікація протоколу обміну даними для підтримки нових команд:

load\_sprite(index, width, height, data)

show\_sprite(index, x, y)

2. Внести виправлення в код емулятора дисплейного модуля для підтримки нових команд (див. п1).

3. Ознайомитися зі змінами в інтерфейсі GraphicsLib.h

4. Внести виправлення в код реалізації інтерфейсу клієнта GrpahicsLib.h для підтримки нових команд (див. п1).

Хід Роботи:

Код:

Клієнт:

Load Sprite:

void loadSprite(uint\_least8\_t index, int\_least16\_t width, int\_least16\_t height, char\* data) override

{

int k = 0;

byte widthArr[] = { width & 0x00FF, width >> 8 };

byte heightArr[] = { height & 0x00FF, height >> 8 };

vector<byte> v;

v.resize(0);

v.push\_back(17);

v.push\_back(index);

v.push\_back(widthArr[0]);

v.push\_back(widthArr[1]);

v.push\_back(heightArr[0]);

v.push\_back(heightArr[1]);

for (int i = 0; i < width \* height \* 3; i++)

{

v.push\_back(data[i]);

}

SendMessageServer(v, v.size());

ListenToServerString();

}

Show Sprite:

void showSprite(uint\_least8\_t index, uint\_least16\_t x, uint\_least16\_t y) override

{

vector<byte> v;

v.resize(0);

v.push\_back(18);

byte indexArr[] = { index & 0x00FF, index >> 8 };

byte xArr[] = { x & 0x00FF, x >> 8 };

byte yArr[] = { y & 0x00FF, y >> 8 };

v.push\_back(indexArr[0]);

v.push\_back(indexArr[1]);

v.push\_back(xArr[0]);

v.push\_back(xArr[1]);

v.push\_back(yArr[0]);

v.push\_back(yArr[1]);

SendMessageServer(v, v.size());

ListenToServerString();

}

Server:

Load Sprite:

case 17:

Console.WriteLine("Command: Upload Image");

ImageUploader(RecievedData);

text = $"Image Uploaded";

Invoke((MethodInvoker)delegate { listBox1.Items.Add(Text = text); });

sendMessage = Encoding.ASCII.GetBytes(text);

server.Send(sendMessage, sendMessage.Length, localEP);

break;Image Uploader:

public static void ImageUploader(byte[] RecievedData)

{

Bitmap bitmap;

int val1place = 2, index, width, height;

byte[] transfer;

index = RecievedData[1];

transfer = new byte[2];

Array.Copy(RecievedData, val1place, transfer, 0, transfer.Length);

width = BitConverter.ToInt16(transfer, 0);

val1place += 2;

Array.Copy(RecievedData, val1place, transfer, 0, transfer.Length);

height = BitConverter.ToInt16(transfer, 0);

transfer = new byte[3];

bitmap = new Bitmap(width, height);

for (int i = 0; i < width; i++)

{

for (int j = 0; j < height; j++)

{

Array.Copy(RecievedData, val1place, transfer, 0, transfer.Length);

bitmap.SetPixel(i, j, Color.FromArgb(transfer[2], transfer[0], transfer[1]));

val1place += 3;

}

}

UploadedImages[index] = bitmap;

}

Show Sprite

case 18:

Console.WriteLine("Command: Draw Image From List");

GetImage(RecievedData, out val1, out val2, out img);

text = $"Command: Draw Image From List, x: {val1} , y: {val2}";

Invoke((MethodInvoker)delegate { listBox1.Items.Add(Text = text); });

DrawImage(command, val1, val2, img);

sendMessage = Encoding.ASCII.GetBytes(text);

server.Send(sendMessage, sendMessage.Length, localEP);

break;

GetImage

public static void GetImage(byte[] RecievedData, out Int16 x, out Int16 y, out System.Drawing.Image img)

{

int val1place = 1, index;

byte[] transfer;

transfer = new byte[2];

Array.Copy(RecievedData, val1place, transfer, 0, transfer.Length);

index = BitConverter.ToInt16(transfer, 0);

val1place += 2;

Array.Copy(RecievedData, val1place, transfer, 0, transfer.Length);

x = BitConverter.ToInt16(transfer, 0);

val1place += 2;

Array.Copy(RecievedData, val1place, transfer, 0, transfer.Length);

y = BitConverter.ToInt16(transfer, 0);

img = UploadedImages[index];

}

DrawImage:

public void DrawImage(byte command, short x1, short y1, System.Drawing.Image img)

{

var Figure = new Figure("Image", x1, y1, img);

Figures.Add(Figure);

Invalidate();

}







