Карпенко О. КН20-1

**ЛАБОРАТОРНА РОБОТА 6**

Удосконалення програм емулятора дисплейного модуля і клієнта.

Етапи виконання лабораторної роботи:

1. Розширити специфікацію протоколу обміну даними для підтримки просунутого набору команд :

set orientation: orientation //(0=0, 1=90, 2=180, 3=270)

get width:

get height:

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

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

4. Опис попередніх команд в специфікації не міняти.

Хід Роботи:

Код:

Client:

GetWidth

int\_least16\_t getWidth(void)

{

vector<byte> v;

v.resize(0);

v.push\_back(15);

SendMessageServer(v, v.size());

char\* b = ListenToServerBytes();

memcpy(&width, b, sizeof(width));

return width;

};

GetHeight

int\_least16\_t getHeight(void)

{

vector<byte> v;

v.resize(0);

v.push\_back(16);

SendMessageServer(v, v.size());

char\* b = ListenToServerBytes();

memcpy(&height, b, sizeof(height));

return height;

};

SetOrientation:

void SetRotation(int\_least16\_t rot)

{

vector<byte> v;

v.resize(0);

v.push\_back(14);

byte rotArr[] = { rot & 0x00FF, rot >> 8};

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

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

SendMessageServer(v, v.size());

ListenToServerString();

}

ListenToServerBytes:

char\* ListenToServerBytes()

{

char buffer[10000];

sockaddr\_in serveraddr;

serveraddr.sin\_family = AF\_INET;

serveraddr.sin\_port = htons(port);

inet\_pton(AF\_INET, ipaddr.c\_str(), &serveraddr.sin\_addr);

int receiveOk = recvfrom(out, (char\*)buffer, 1024, 0, NULL, NULL);

buffer[receiveOk] = '\0';

return buffer;

}

Server:

case 14:

RotateImage(RecievedData, out val1);

text = $"Command: Set Orientation; {val1} degrees";

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

sendMessage = Encoding.ASCII.GetBytes(text);

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

break;

case 15:

val1 = Convert.ToInt16(this.Width);

text = $"Command: Get Width; Width: {val1} px";

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

bytes = new List<byte>();

bytes.AddRange(BitConverter.GetBytes(val1));

sendMessage = bytes.ToArray();

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

break;

case 16:

val1 = Convert.ToInt16(this.Height);

Console.WriteLine($"Height: {val1} px");

text = $"Command: Get Height; Height: {val1} px";

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

bytes = new List<byte>();

bytes.AddRange(BitConverter.GetBytes(val1));

sendMessage = bytes.ToArray();

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

break;

RotateImage:

public void RotateImage(byte[] RecievedData, out Int16 Orientation)

{

byte[] transfer;

int val1place = 1;

transfer = new byte[2];

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

Orientation = BitConverter.ToInt16(transfer, 0);

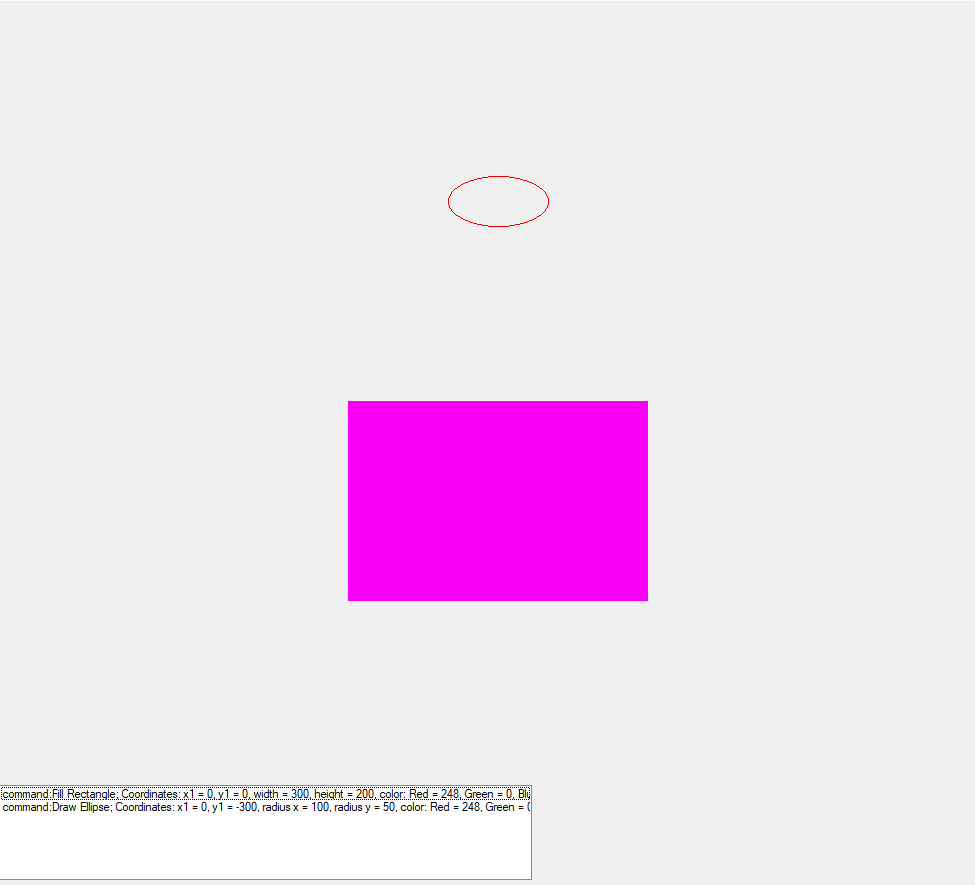
rotation = (int)Orientation;

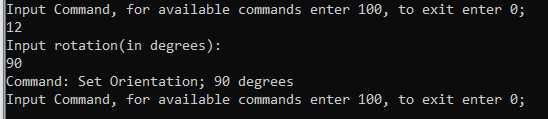
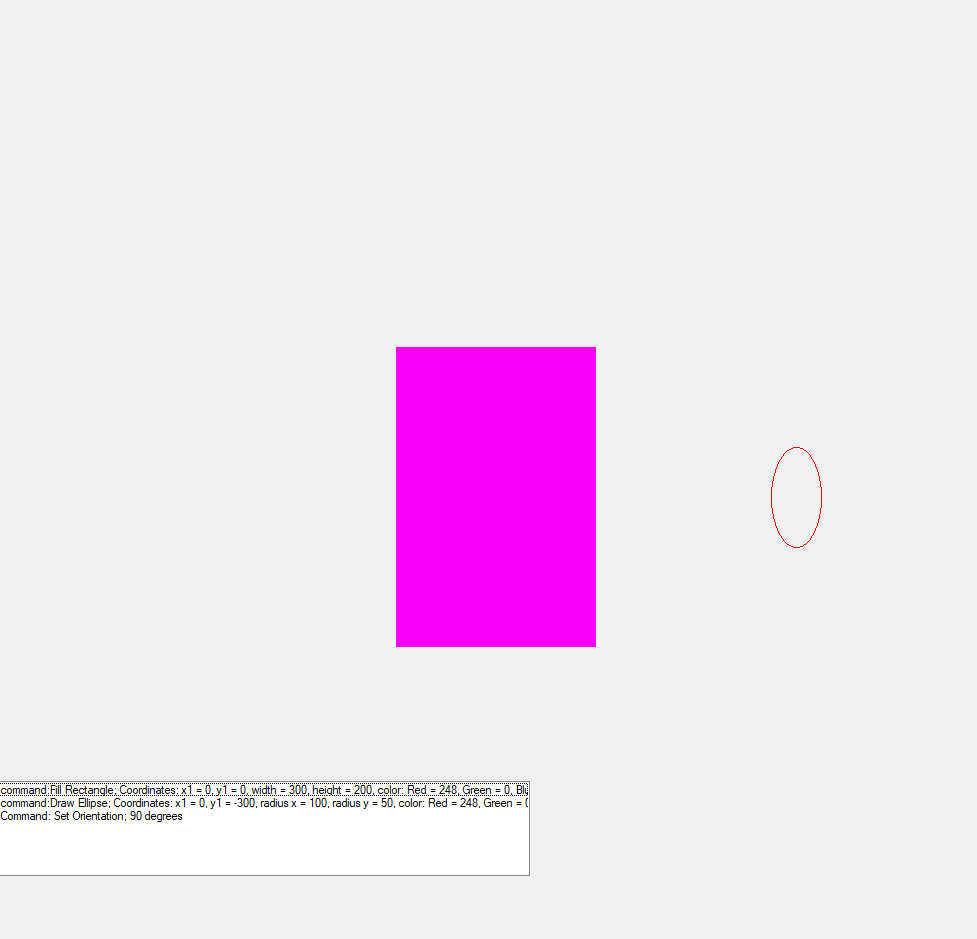
Invalidate();

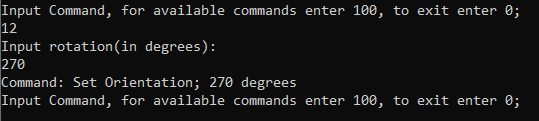
}

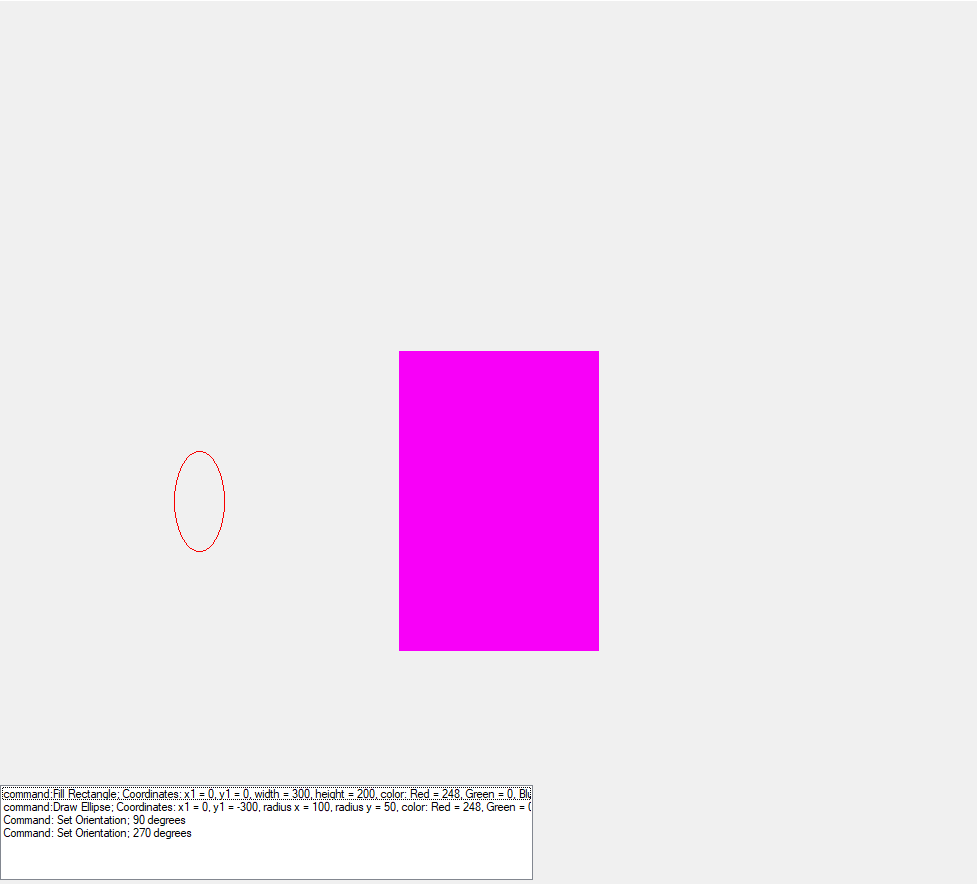
Set Orientation:

Створив 2 фігури для наглядного показу:







GetWidth:

