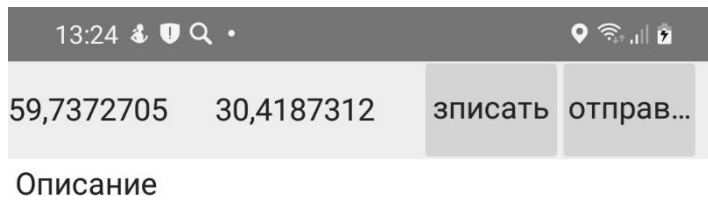


Спутниковые программы

1. Тахеосъемка



13:24

59,7372705 30,4187312

зписать отправ...

Описание

6627801,0 354977,9 Описание
6627889,5 354900,5 Описание



III O <

Кнопка записать записывает координаты и описание во второй мемо.

Кнопка отправить отправляет через выбранную программу.

procedure Preobr(Aa, Ab, Ea, Eb, B, Lat, H, dx, dy, dz, wx, wy, wz, mm: double);

var

a, e2, da, de2, M, n: double;

```

begin
  a := (Ab + Aa) / 2;
  e2 := (sqr(Eb) + sqr(Ea)) / 2;
  da := Ab - Aa;
  de2 := sqr(Eb) - sqr(Ea);
  M := a * (1 - e2) * Power((1 - e2 * sqr(sin(B))), -3 / 2);
  n := a * Power((1 - e2 * sqr(sin(B))), -1 / 2);
  dB := p / (M + H) * ((n / a) * e2 * sin(B) * cos(B) * da +
    (sqr(n) / sqr(a) + 1) * n * sin(B) * cos(B) * (de2 / 2) -
    (dx * cos(L) + dy * sin(L)) * sin(B) + dz * cos(B)) - wx * sin(L) *
    (1 + e2 * cos(2 * B)) + wy * cos(L) * (1 + e2 * cos(2 * B)) - p * mm * e2 *
    sin(B) * cos(B);
  dL := (p / ((n + H) * cos(B))) * (-dx * sin(L) + dy * cos(L)) + tan(B) *
    (1 - e2) * (wx * cos(L) + wy * sin(L)) - wz;
  dH := (-a / n) * da + n * sqr(sin(B)) * (de2 / 2) + (dx * cos(L) + dy * sin(L)
    ) * cos(B) + dz * sin(B) - n * e2 * sin(B) * cos(B) *
    ((wx / p) * sin(L) - (wy / p) * cos(L)) + (sqr(a) / n + H) * mm;
end;

// из БГС в СК
procedure TForm1.WGS_SK;
begin
  Lat := DegToRad(Lat);
  Lon := DegToRad(Lon);
  Preobr(6378137, 6378136, 1 / 298.257223563, 1 / 298.25784, Lat, Lon, Heig,
    1.08, 0.27, 0.9, 0, 0, 0.16, 0.12E-6); // из БГС в ПЗ
  dB := dB / 3600;
  dL := dL / 3600;
  Lat := RadToDeg(Lat) + dB;
  Lon := RadToDeg(Lon) + dL;
  Heig := LocationSensor1.Sensor.Altitude; // Heig + dH;
  Lat := DegToRad(Lat);
  Lon := DegToRad(Lon);
  Preobr(6378136, 6378245, 1 / 298.25784, 1 / 298.3, Lat, Lon, Heig, -25.9,
    130.94, 81.76, 0, 0, 0, 0); // из ПЗ в СК 95
  dB := dB / 3600;
  dL := dL / 3600;
  Lat := RadToDeg(Lat) + dB;
  Lon := RadToDeg(Lon) + dL;
  Heig := Heig + dH;
  B := Lat;
  L := Lon;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  F1 := LocationSensor1.Sensor.Latitude;
  L1 := LocationSensor1.Sensor.Longitude;
  WGS_SK(F1, L1, LocationSensor1.Sensor.Altitude);
  SK_Pr(B, L);
end;

```

```

x1 := x;
y1 := y;
Memo2.Lines.Add(FloatToStrF(x1, ffFixed, 10, 1)+' '+
FloatToStrF(y1, ffFixed, 10, 1)+' '+
Memo1.Text);
end;

procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
const OldLocation, NewLocation: TLocationCoord2D);
begin
Label1.Text:=NewLocation.Latitude.ToString;
Label2.Text:=NewLocation.Longitude.ToString;
end;

procedure TForm1.ShowShareSheetAction1BeforeExecute(Sender: TObject);
begin
ShowShareSheetAction1.TextMessage:=Memo2.Text;
end;

// в прямоугор
procedure TForm1.SK_Pr;
var
n, cos_sqr_B, a0, a4, a6, a3, a5, B_sek, N_Z, L0: Extended;
begin
Lat := DegToRad(Lat); // Широта в радианах
cos_sqr_B := sqr(cos(Lat));
n := 6399698.902 - (21562.267 - (108.973 - 0.612 * cos_sqr_B) * cos_sqr_B) *
cos_sqr_B;
a0 := 32140.404 - (135.3302 - (0.7092 - 0.004 * cos_sqr_B) * cos_sqr_B) *
cos_sqr_B;
a4 := (0.25 + 0.00252 * cos_sqr_B) * cos_sqr_B - 0.04166;
a6 := (0.166 * cos_sqr_B - 0.084) * cos_sqr_B;
a3 := (0.3333333 + 0.001123 * cos_sqr_B) * cos_sqr_B - 0.1666667;
a5 := 0.0083 - (0.1667 - (0.1968 + 0.004 * cos_sqr_B) * cos_sqr_B) *
cos_sqr_B;
N_Z := Round((Lon + 3) / 6);
L0 := 6 * N_Z - 3;
Lon := (Lon - L0) * 3600;
Lon := Lon / ro;
B_sek := RadToDeg(Lat) * 3600;
x := 6367558.4969 * B_sek / ro - (a0 - (0.5 + (a4 + a6 * sqr(Lon)) * sqr(Lon))
* sqr(Lon) * n) * sin(Lat) * cos(Lat);
y := (1 + (a3 + a5 * sqr(Lon)) * sqr(Lon)) * Lon * n * cos(Lat);
y := 500000 + y;
end;

```

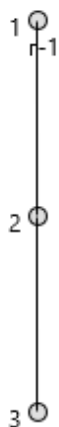
2. Промер

Промер

Журнал	Планшет				
1	1	100	100	Добавить	Удалить
1	1	100	100		
1	2	100	200		
1	3	100	300		

Промер

Журнал	Планшет	
--------	---------	--



Координаты заносятся вручную и обрисовываются на планшете. При закрытии программы сохраняются в четырех текстовых файлах. При запуске программы из этих файлов считывается информация и рисует на планшете. Кнопка добавить добавляет данные из введенных в поля. Кнопка удалить удаляет отдельно в каждом поле.

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
begin
```

```
Listbox1.Items.Add(Edit1.text);
```

```
Listbox2.Items.Add(Edit2.text);
```

```
Listbox3.Items.Add(Edit3.text);
```

```
Listbox4.Items.Add(Edit4.text);
```

```
//circle
```

```
C := Tcircle.Create(Image1);
```

```
C.Parent := Image1;
```

```
C.Position.X := Edit3.text.ToSingle;
```

```
C.Position.Y := Edit4.text.ToSingle;
```

```
C.Width := 10;
```

```
C.Height := 10;
```

```
//number place
```

```
L := TLabel.Create(Image1);
```

```
L.Parent := Image1;
```

```
L.Position.X := Edit3.text.ToSingle - 10;
```

```

L.Position.Y := Edit4.text.ToSingle;
L.text := Edit2.text;
//number galse
L := TLabel.Create(Image1);
L.Parent := Image1;
L.Position.X := Edit3.text.ToSingle;
L.Position.Y := Edit4.text.ToSingle + 10;
if Edit2.Text='1' then L.text := 'r-'+Edit1.text;
// линия
if Edit2.Text <> '1' then
begin
dx := ListBox3.Items[ListBox3.Count - 1].ToExtended - ListBox3.Items[ListBox3.Count -
2].ToExtended;
dy := ListBox4.Items[ListBox4.Count - 1].ToExtended - ListBox4.Items[ListBox4.Count -
2].ToExtended;
s := Math.Hypot(dx, dy);
u := RadToDeg(arsin(dx / s));
if -dy < 0 then
u := 180 - u;
if u < 0 then
u := 360 + u;
Lin := Tline.Create(Image1);
Lin.Parent := Image1;
Lin.Position.x := Edit3.Text.ToSingle+5;
Lin.Position.y := Edit4.Text.ToSingle+5;
Lin.RotationCenter.x := 0;
Lin.RotationCenter.y := 0;
Lin.RotationAngle := u;
Lin.Width := 1;
Lin.Height := s;
end;
end;

```

```

procedure TForm1.Button2Click(Sender: TObject);
begin
ListBox1.Items.Delete(ListBox1.ItemIndex);
ListBox2.Items.Delete(ListBox2.ItemIndex);
ListBox3.Items.Delete(ListBox3.ItemIndex);
ListBox4.Items.Delete(ListBox4.ItemIndex);
end;

```

```

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
ListBox1.Items.SaveToFile('1.txt');
ListBox2.Items.SaveToFile('2.txt');
ListBox3.Items.SaveToFile('3.txt');
ListBox4.Items.SaveToFile('4.txt');
end;

```

```

procedure TForm1.FormCreate(Sender: TObject);

```

```

var i:integer;
begin
try
ListBox1.Items.LoadFromFile('1.txt');
ListBox2.Items.LoadFromFile('2.txt');
ListBox3.Items.LoadFromFile('3.txt');
ListBox4.Items.LoadFromFile('4.txt');
for i := 0 to ListBox1.Items.Count-1 do
begin
//circle
C := Tcircle.Create(Image1);
C.Parent := Image1;
C.Position.X := ListBox3.Items[i].ToSingle;
C.Position.Y := ListBox4.Items[i].ToSingle;
C.Width := 10;
C.Height := 10;
//number place
L := TLabel.Create(Image1);
L.Parent := Image1;
L.Position.X := ListBox3.items[i].ToSingle - 10;
L.Position.Y := ListBox4.items[i].ToSingle;
L.text := ListBox2.Items[i];
//number galse
L := TLabel.Create(Image1);
L.Parent := Image1;
L.Position.X := ListBox3.Items[i].ToSingle;
L.Position.Y := ListBox4.Items[i].ToSingle + 10;
if ListBox2.Items[i]='1' then L.text := 'r-'+ListBox1.Items[i];
// линия
if ListBox2.Items[i] <> '1' then
begin
dx := ListBox3.Items[ListBox3.Count - 1].ToExtended - ListBox3.Items[ListBox3.Count -
2].ToExtended;
dy := ListBox4.Items[ListBox4.Count - 1].ToExtended - ListBox4.Items[ListBox4.Count -
2].ToExtended;
s := Math.Hypot(dx, dy);
u := RadToDeg(arsin(dx / s));
if -dy < 0 then
u := 180 - u;
if u < 0 then
u := 360 + u;
Lin := Tline.Create(Image1);
Lin.Parent := Image1;
Lin.Position.x := ListBox3.Items[i].ToSingle+5;
Lin.Position.y := ListBox4.Items[i].ToSingle+5;
Lin.RotationCenter.x := 0;
Lin.RotationCenter.y := 0;
Lin.RotationAngle := u;
Lin.Width := 1;
Lin.Height := s;


```

```
end;  
end;  
finally
```

```
end;
```

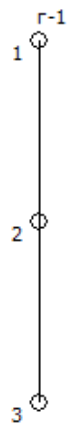
3. Промер.

Лучше предыдущей написан на библиотеке vsl. Создание графических примитивов поддерживается на канве компонента Image. На fairmanke не получилось с линиями. А здесь поддерживается из начальной точки в конечную.

 Промер

— □ ×

Журнал Планшет



Промер

Журнал Планшет

1	1	150	150	Нанести	Очистить	Очистить все
1	1	150	150			
1	2	150	250			
1	3	150	350			

```

procedure TForm1.Button1Click(Sender: TObject);
var
  x1, y1, x2, y2, xLine1, yLine1, xLine2, yLine2: integer;
begin
  x1 := StrToInt(Edit3.Text);
  y1 := StrToInt(Edit4.Text);
  x2 := StrToInt(Edit3.Text) + 10;
  y2 := StrToInt(Edit4.Text) + 10;
  Image1.Canvas.Ellipse(x1-5, y1-5, x2-5, y2-5);
  Image1.Canvas.TextOut(x1 - 15, y1, Edit2.Text);
  if Edit2.Text = '1' then
    Image1.Canvas.TextOut(x1, y1 - 20, 'r-' + Edit1.Text);
  ListBox1.Items.Add(Edit1.Text);
  ListBox2.Items.Add(Edit2.Text);
  ListBox3.Items.Add(Edit3.Text);
  ListBox4.Items.Add(Edit4.Text);
  xLine2 := StrToInt(Edit3.Text);
  yLine2 := StrToInt(Edit4.Text);
  if Edit2.Text <> '1' then
    begin
      xLine1 := ListBox3.Items[ListBox3.Items.Count - 2].ToInteger;
      yLine1 := ListBox4.Items[ListBox4.Items.Count - 2].ToInteger;
      Image1.Canvas.MoveTo(xLine1, yLine1);
      Image1.Canvas.LineTo(xLine2, yLine2);
    end;
end;

```

```
Edit2.Text:=IntToStr(StrToInt(Edit2.Text)+1);  
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);  
begin  
  ListBox1.Items[ListBox1.ItemIndex];  
  ListBox2.Items[ListBox2.ItemIndex];  
  ListBox3.Items[ListBox3.ItemIndex];  
  ListBox4.Items[ListBox4.ItemIndex];  
end;
```

```
procedure TForm1.Button3Click(Sender: TObject);  
begin  
  ListBox1.Clear;  
  ListBox2.Clear;  
  ListBox3.Clear;  
  ListBox4.Clear;  
end;
```

```
procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);  
begin  
  ListBox1.Items.SaveToFile('1.txt');  
  ListBox2.Items.SaveToFile('2.txt');  
  ListBox3.Items.SaveToFile('3.txt');  
  ListBox4.Items.SaveToFile('4.txt');  
end;
```

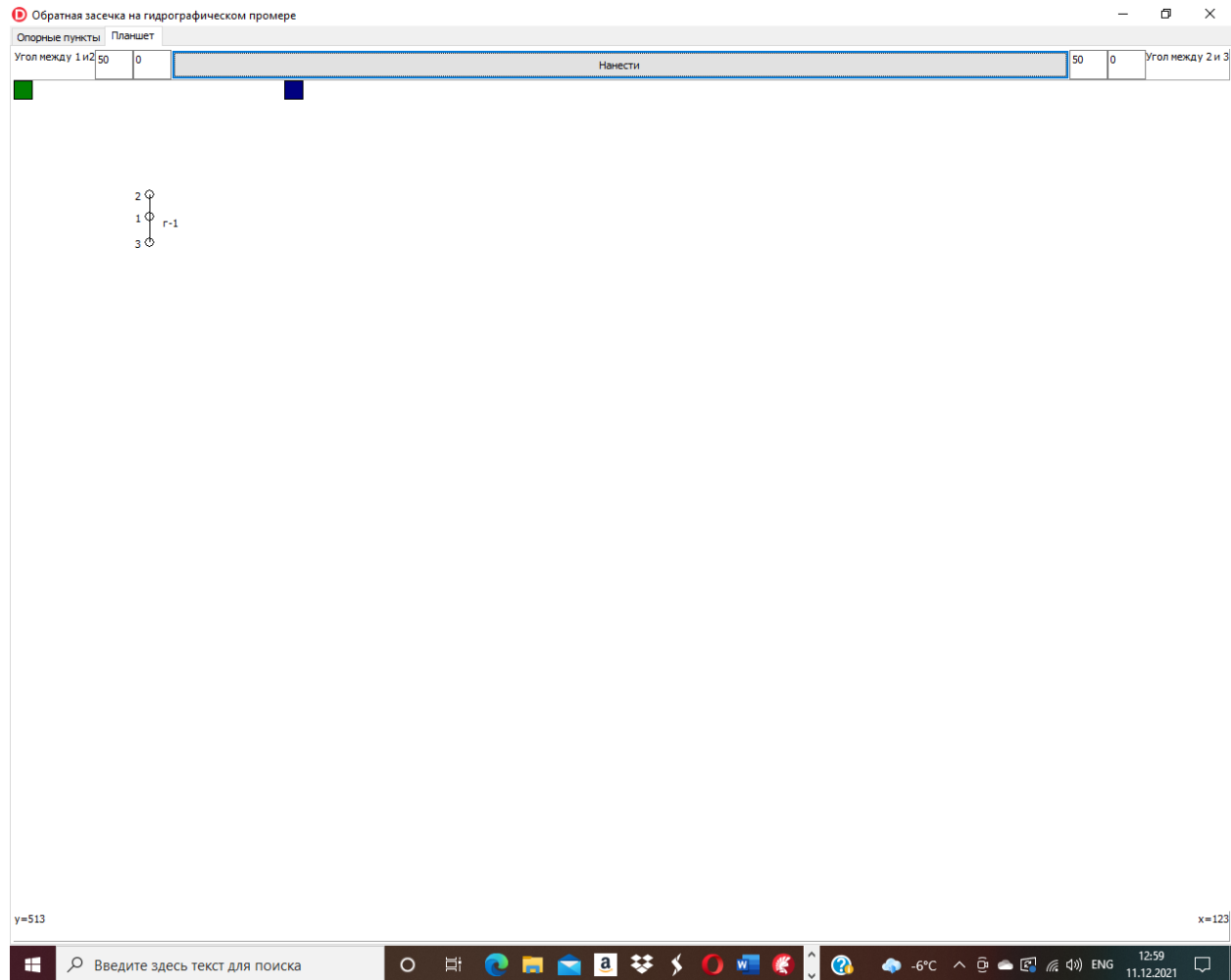
```
procedure TForm1.FormShow(Sender: TObject);  
var  
  i,x1, y1, x2, y2, xLine1, yLine1, xLine2, yLine2: integer;  
begin  
  ListBox1.Items.LoadFromFile('1.txt');  
  ListBox2.Items.LoadFromFile('2.txt');  
  ListBox3.Items.LoadFromFile('3.txt');  
  ListBox4.Items.LoadFromFile('4.txt');  
  for i := 0 to ListBox1.Items.Count-1 do  
    begin  
      x1:=ListBox3.Items[i].ToInteger;  
      y1:=ListBox4.Items[i].ToInteger;  
      x2:=x1+10;  
      y2:=y1+10;  
      Image1.Canvas.Ellipse(x1-5,y1-5,x2-5,y2-5);  
      Image1.Canvas.TextOut(x1-15,y1,ListBox2.Items[i]);  
      if ListBox2.Items[i]='1' then  
        Image1.Canvas.TextOut(x1,y1-20,'r-'+ListBox1.Items[i]);  
      if ListBox2.Items[i]<>'1' then  
        begin  
          xLine2:=ListBox3.Items[i].ToInteger;  
          yLine2:=ListBox4.Items[i].ToInteger;  
          xLine1:=ListBox3.Items[i-1].ToInteger;
```

```

yLine1:=ListBox4.Items[i-1].ToInteger;
Image1.Canvas.MoveTo(xLine1, yLine1);
Image1.Canvas.LineTo(xLine2, yLine2);
end;
end;
end;

```

4. Прямая засечка.



Обратная засечка на гидрографическом промере

Опорные пункты Планшет

Пункт №1 Пункт №2 Пункт №3 Пункт №4

Номер галса Номер определения

Очистить

1	1	142	142
---	---	-----	-----

Windows taskbar: Введите здесь текст для поиска, -6°C, 12:59, 11.12.2021

Координирование промера по прямой засечке.

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
  xp, yp, x1, y1, x2, y2, cotan1, cotan2, x1line, y1line, tgA1p, b1, b2, a2, a3, a1p: single;
```

```
begin
```

```
  x1 := StrToInt(Edit2.Text);
```

```
  y1 := StrToInt(Edit1.Text);
```

```
  x2 := StrToInt(Edit4.Text);
```

```
  y2 := StrToInt(Edit3.Text);
```

```
  b1 := StrToInt(Edit5.Text) + StrToFloat(Edit6.Text) / 60;
```

```
  b2 := StrToInt(Edit7.Text) + StrToFloat(Edit8.Text) / 60;
```

```
  cotan1 := cotan(DegToRad(b1));
```

```
  cotan2 := cotan(DegToRad(b2));
```

```
  xp := (x1 * cotan2 + x2 * cotan1 - y1 + y2) / (cotan1 + cotan2);
```

```
  yp := (y1 * cotan2 + y2 * cotan1 + x1 - x2) / (cotan1 + cotan2);
```

```
  ListBox1.Items.Add(Edit9.Text);
```

```
  ListBox2.Items.Add(Edit10.Text);
```

```
  ListBox3.Items.Add(FloatToStr(round(xp)));
```

```
  ListBox4.Items.Add(FloatToStr(round(yp)));
```

```
  Image1.Canvas.Ellipse(round(yp) - 5, round(xp) - 5, round(yp) + 5, round(xp) + 5);
```

```
  Image1.Canvas.TextOut(round(yp) - 15, round(xp) - 5, Edit10.Text);
```

```
  if Edit10.Text = '1' then
```

```
    Image1.Canvas.TextOut(round(yp) + 15, round(xp), 'r-' + Edit9.Text)
```

```

else
begin
    x1line := StrToFloat(ListBox4.Items[ListBox4.Items.Count - 2]);
    y1line := StrToFloat(ListBox3.Items[ListBox3.Items.Count - 2]);
    Image1.Canvas.MoveTo(round(x1line), round(y1line));
    Image1.Canvas.LineTo(round(yp), round(xp));
end;
Edit10.Text := IntToStr(StrToInt(Edit10.Text) + 1);
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
    ListBox1.Items.Clear;
    ListBox2.Items.Clear;
    ListBox3.Items.Clear;
    ListBox4.Items.Clear;
end;

procedure TForm1.Edit1Change(Sender: TObject);
begin
    Shape1.Left := StrToInt(Edit1.Text);
end;

procedure TForm1.Edit2Change(Sender: TObject);
begin
    Shape1.Top := StrToInt(Edit2.Text);
end;

procedure TForm1.Edit3Change(Sender: TObject);
begin
    Shape2.Left := StrToInt(Edit3.Text);
end;

procedure TForm1.Edit4Change(Sender: TObject);
begin
    Shape2.Top := StrToInt(Edit4.Text);
end;

procedure TForm1.FormShow(Sender: TObject);
begin
    Shape2.Left := Round(Image1.Width/2);
    Shape2.Top := Image1.Top;
    Shape1.Top := Image1.Top;
    Shape1.Left := Image1.Left;
    Edit3.Text := IntToStr(Shape2.Left);
end;

procedure TForm1.Image1MouseMove(Sender: TObject; Shift: TShiftState;
    X, Y: Integer);
begin

```

```
Label7.Caption := 'y=' + X.ToString;
Label8.Caption := 'x=' + Y.ToString;
end;
```

```
end.
```

5. Обратная засечка.

Промер по обратной засечке

Журнал

Планшет

Угол 1

45

0

Угол 2

45

0

1

○

г-1

148

Нанести

272

Промер по обратной засечке

Журнал

Планшет

Пункт 1

308

Пункт 2

264

308

Пункт 3

508

308

Галс 1

ОМС 2

Западная

300000

Северная

6000000

1

1

55

274

300055

5999971

Вычисляет точки галса и рисует на планшете.

```

procedure TForm1.Button1Click(Sender: TObject);
var
  x1, y1, x2, y2, x3, y3, cotan1, cotan2, b1, b2, tanA1, a1, a2, a3, xp, yp1,
  yp2, x1line, y1line: real;
begin
  x1 := strtofloat(Edit4.Text);
  y1 := strtofloat(Edit3.Text);
  x2 := strtofloat(Edit8.Text);
  y2 := strtofloat(Edit7.Text);
  x3 := strtofloat(Edit10.Text);
  y3 := strtofloat(Edit9.Text);
  b1 := strtofloat(Edit1.Text) + strtofloat(Edit2.Text) / 60;
  b2 := strtofloat(Edit5.Text) + strtofloat(Edit6.Text) / 60;
  b2 := b1 + b2;
  cotan1 := cotan(DegToRad(b1));
  cotan2 := cotan(DegToRad(b2));
  tanA1 := ((y2 - y1) * cotan1 + (y1 - y3) * cotan2 - x2 + x3) /
    ((x2 - x1) * cotan1 + (x1 - x3) * cotan2 + y2 - y3);
  a1 := RadToDeg(arctan(tanA1));
  a2 := a1 + b1;
  // a3 := a1 + b2;
  xp := (x1 * tanA1 - x2 * tan(DegToRad(a2)) + y2 - y1) /
    (tanA1 - tan(DegToRad(a2)));
  yp1 := y1 + (xp - x1) * tanA1;
  yp2 := y2 + (xp - x2) * tan(DegToRad(a2));
  ListBox1.Items.Add(Edit11.Text);
  ListBox2.Items.Add(Edit12.Text);
  ListBox3.Items.Add(FloatToStr(round(xp)));
  ListBox4.Items.Add(FloatToStr(round(yp1)));
  ListBox5.Items.Add(FloatToStr(round(xp+StrToInt(Edit13.Text))));
  ListBox6.Items.Add(FloatToStr(round(yp1-H+StrToInt(Edit14.Text))));
  Image1.Canvas.Ellipse(round(yp1) - 5, round(xp) - 5, round(yp1) + 5,
    round(xp) + 5);
  Image1.Canvas.TextOut(round(yp1) - 15, round(xp) - 5, Edit12.Text);
  if Edit12.Text = '1' then
    Image1.Canvas.TextOut(round(yp1) + 15, round(xp), 'r-' + Edit11.Text)
  else
    begin
      x1line := StrToFloat(ListBox4.Items[ListBox4.Items.Count - 2]);
      y1line := StrToFloat(ListBox3.Items[ListBox3.Items.Count - 2]);
      Image1.Canvas.MoveTo(round(x1line), round(y1line));
      Image1.Canvas.LineTo(round(yp1), round(xp));
    end;
  Edit12.Text := intToStr(StrToInt(Edit12.Text) + 1);
end;

procedure TForm1.Edit10Change(Sender: TObject);
begin
  Shape3.Top := StrToInt(Edit10.Text);
end;

```

```
procedure TForm1.Edit3Change(Sender: TObject);
begin
  Shape1.Left := StrToInt(Edit1.Text);
end;
```

```
procedure TForm1.Edit4Change(Sender: TObject);
begin
  Shape1.Top := StrToInt(Edit4.Text);
end;
```

```
procedure TForm1.Edit7Change(Sender: TObject);
begin
  Shape2.Left := StrToInt(Edit7.Text);
end;
```

```
procedure TForm1.Edit8Change(Sender: TObject);
begin
  Shape2.Top := StrToInt(Edit8.Text);
end;
```

```
procedure TForm1.Edit9Change(Sender: TObject);
begin
  Shape3.Left := StrToInt(Edit9.Text);
end;
```

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  Shape1.Left := 0;
  Shape2.Left := round(Image1.Width / 2);
  Shape3.Left := Image1.Width - Shape3.Width;
  Shape1.Top := Image1.Top + Image1.Height - Shape1.Height;
  Shape2.Top := Image1.Top + Image1.Height - Shape2.Height;
  Shape3.Top := Image1.Top + Image1.Height - Shape3.Height;
  Edit3.Text := intToStr(Shape1.Left);
  Edit7.Text := intToStr(Shape2.Left);
  Edit9.Text := intToStr(Shape3.Left);
  Edit4.Text := intToStr(Shape1.Top);
  Edit8.Text := intToStr(Shape2.Top);
  Edit10.Text := intToStr(Shape3.Top);
  H:=Image1.Height;
end;
```

```
procedure TForm1.Image1MouseMove(Sender: TObject; Shift: TShiftState;
  X, Y: Integer);
begin
  Label1.Caption := intToStr(X);
  Label2.Caption := intToStr(Y);
end;
```

6. Прямая засечка.

прямая засечка на гидрографическом промере

Опорные пункты Планшет

Угол между 1 и 2 55 0 45 0 Угол между 2 и 1

у=99 x=16

1
2

г-1

прямая засечка на гидрографическом промере

Опорные пункты Планшет

Пункт №1 100 50 Пункт №2 283 50

Номер галса 1 Номер места 3

1	1	141,5	191,5
1	2	157,6339263916	175,3660888671

Координирование промера по прямой засечке в местной системе координат от северо западного угла планшета.

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
xp, yp, x1, y1, x2, y2, cotan1, cotan2, x1line, y1line, tgA1p, b1, b2, a2, a3,
```

```

    a1p, yk, xk: single;
begin
    x1 := StrToInt(Edit2.Text);
    y1 := StrToInt(Edit1.Text);
    x2 := StrToInt(Edit4.Text);
    y2 := StrToInt(Edit3.Text);
    b1 := StrToInt(Edit5.Text) + StrToFloat(Edit6.Text) / 60;
    b2 := StrToInt(Edit7.Text) + StrToFloat(Edit8.Text) / 60;
    cotan1 := cotan(DegToRad(b1));
    cotan2 := cotan(DegToRad(b2));
    xp := (x1 * cotan2 + x2 * cotan1 - y1 + y2) / (cotan1 + cotan2);
    yp := (y1 * cotan2 + y2 * cotan1 + x1 - x2) / (cotan1 + cotan2);
    ListBox1.Items.Add(Edit9.Text);
    ListBox2.Items.Add(Edit10.Text);
    xk := xp;
    yk := yp;
    ListBox3.Items.Add(FloatToStr(xk));
    ListBox4.Items.Add(FloatToStr(yk));
    Image1.Canvas.Ellipse(round(yk) - 5, round(xk) - 5, round(yk) + 5,
        round(xk) + 5);
    Image1.Canvas.TextOut(round(yk) - 15, round(xk) - 5, Edit10.Text);
    if Edit10.Text = '1' then
        Image1.Canvas.TextOut(round(yk) + 15, round(xk), 'r-' + Edit9.Text)
    else
        begin
            x1line := StrToFloat(ListBox3.Items[ListBox3.Items.Count - 2]);
            y1line := StrToFloat(ListBox4.Items[ListBox4.Items.Count - 2]);
            Image1.Canvas.MoveTo(round(y1line), round(x1line));
            Image1.Canvas.LineTo(round(yk), round(xk));
        end;
    Edit10.Text := IntToStr(StrToInt(Edit10.Text) + 1);
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
    ListBox1.Items.Clear;
    ListBox2.Items.Clear;
    ListBox3.Items.Clear;
    ListBox4.Items.Clear;
end;

procedure TForm1.Edit1Change(Sender: TObject);
begin
    Shape1.Left := StrToInt(Edit1.Text);
end;

procedure TForm1.Edit2Change(Sender: TObject);
begin
    Shape1.Top := StrToInt(Edit2.Text);
end;

```

```
procedure TForm1.Edit3Change(Sender: TObject);  
begin  
    Shape2.Left := StrToInt(Edit3.Text);  
end;
```

```
procedure TForm1.Edit4Change(Sender: TObject);  
begin  
    Shape2.Top := StrToInt(Edit4.Text);  
end;
```

```
procedure TForm1.FormResize(Sender: TObject);  
begin  
    H := Image1.Height;  
end;
```

```
procedure TForm1.FormShow(Sender: TObject);  
begin  
    Shape2.Left := round(Image1.Width / 2);  
    Shape2.Top := Image1.Top;  
    Shape1.Top := Image1.Top;  
    Shape1.Left := Image1.Left;  
    Edit3.Text := IntToStr(Shape2.Left);  
    H := Image1.Height;  
end;
```

```
procedure TForm1.Image1MouseMove(Sender: TObject; Shift: TShiftState;  
    X, Y: Integer);  
begin  
    Label7.Caption := 'y=' + X.ToString;  
    Label8.Caption := 'x=' + Y.ToString;  
end;
```

7. Обратная засечка.

Промер по обратной засечке

Журнал Планшет

Угол 1 15 0 Угол 2 15 0

187 171

Нанести

Промер по обратной засечке

Журнал Планшет

Пункт 1 100 20 Пункт 2 400 20 Пункт 3 800 20

Галс 1 ОМС 5

1	1	352	316
1	2	352	316
1	3	805,4495104487	374,3977845974
1	4	131,2035945208	976,3490191786

В местных координатах от северо-западного угла

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
  x1, y1, x2, y2, x3, y3, cotan1, cotan2, b1, b2, tanA1, a1, a2, a3, xp, yp1,
```

```
  yp2, x1Line, y1Line, xk, yk: real;
```

```
begin
```

```
  y1 := strtoint(Edit3.Text);
```

```

x1 := strtofloat(Edit4.Text);
y2 := strtofloat(Edit7.Text);
x2 := strtofloat(Edit8.Text);
y3 := strtofloat(Edit9.Text);
x3 := strtofloat(Edit10.Text);
b1 := strtofloat(Edit1.Text) + strtofloat(Edit2.Text) / 60;
b2 := strtofloat(Edit5.Text) + strtofloat(Edit6.Text) / 60;
b2 := b1 + b2;
cotan1 := cotan(DegToRad(b1));
cotan2 := cotan(DegToRad(b2));
tanA1 := ((y2 - y1) * cotan1 + (y1 - y3) * cotan2 - x2 + x3) /
  ((x2 - x1) * cotan1 + (x1 - x3) * cotan2 + y2 - y3);
a1 := RadToDeg(arctan(tanA1));
a2 := a1 + b1;
xp := (x1 * tanA1 - x2 * tan(DegToRad(a2)) + y2 - y1) /
  (tanA1 - tan(DegToRad(a2)));
yp1 := y1 + (xp - x1) * tanA1;
yp2 := y2 + (xp - x2) * tan(DegToRad(a2));
ListBox1.Items.Add(Edit11.Text);
ListBox2.Items.Add(Edit12.Text);
xk:= abs(yp1);
yk:=abs(xp);
ListBox3.Items.Add(FloatToStr(xk));
ListBox4.Items.Add(FloatToStr(yk));
Image1.Canvas.Ellipse(round(yk)-5,round(xk)-5,round(yk)+5,round(xk)+5);
Image1.Canvas.TextOut(round(yk) - 15, round(xk) - 5, Edit12.Text);
if Edit12.Text = '1' then
  Image1.Canvas.TextOut(round(yk) + 15, round(xk), 'r-' + Edit11.Text)
else
begin
  x1line := StrToFloat(ListBox3.Items[ListBox3.Items.Count - 2]);
  y1line := StrToFloat(ListBox4.Items[ListBox4.Items.Count - 2]);
  Image1.Canvas.MoveTo(round(y1line), round(x1line));
  Image1.Canvas.LineTo(round(yk), round(xk));
end;
Edit12.Text := intToStr(StrToInt(Edit12.Text) + 1);
end;

procedure TForm1.Edit10Change(Sender: TObject);
begin
  Shape3.Top := StrToInt(Edit10.Text);
end;

procedure TForm1.Edit3Change(Sender: TObject);
begin
  Shape1.Left := StrToInt(Edit3.Text);
end;

procedure TForm1.Edit4Change(Sender: TObject);
begin

```

```

    Shape1.Top := StrToInt(Edit4.Text);
end;

procedure TForm1.Edit7Change(Sender: TObject);
begin
    Shape2.Left := StrToInt(Edit7.Text);
end;

procedure TForm1.Edit8Change(Sender: TObject);
begin
    Shape2.Top := StrToInt(Edit8.Text);
end;

procedure TForm1.Edit9Change(Sender: TObject);
begin
    Shape3.Left := StrToInt(Edit9.Text);
end;

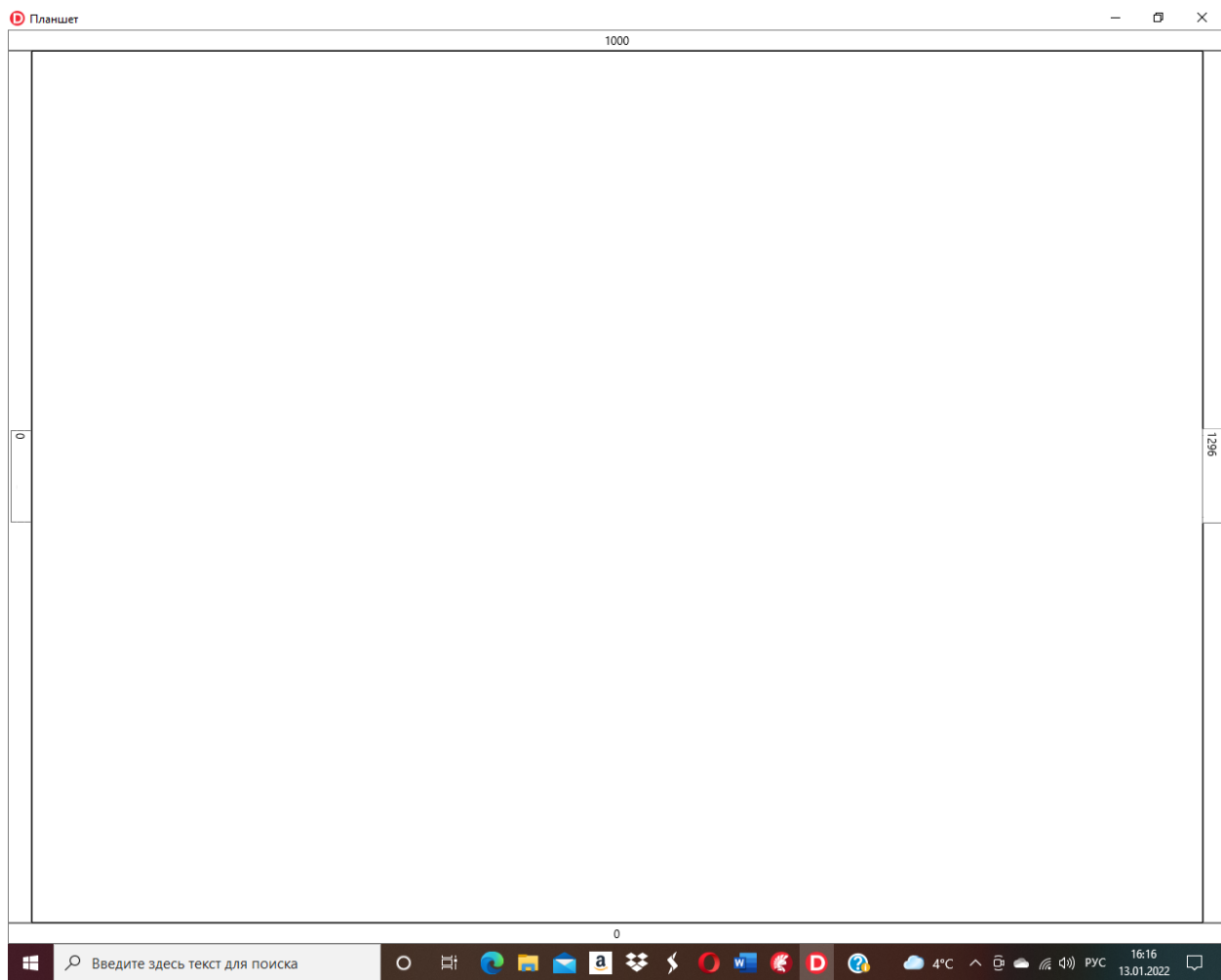
procedure TForm1.FormCreate(Sender: TObject);
begin
    Shape1.Left := 0;
    Shape2.Left := round(Image1.Width / 2);
    Shape3.Left := Image1.Width - Shape3.Width;
    Shape1.Top := Image1.Top + Image1.Height - Shape1.Height;
    Shape2.Top := Image1.Top + Image1.Height - Shape2.Height;
    Shape3.Top := Image1.Top + Image1.Height - Shape3.Height;
    H:=Image1.Height;
end;

procedure TForm1.FormResize(Sender: TObject);
begin
    H:=Image1.Height;
end;

procedure TForm1.Image1MouseMove(Sender: TObject; Shift: TShiftState;
    X, Y: Integer);
begin
    Label1.Caption := intToStr(X);
    Label2.Caption := intToStr(Y);
end;

```

8. Построение планшета Гаусса



Строится планшет в проекции Гаусса-Крюгера

Пересчет происходит при изменении размеров формы.

```
procedure TForm1.FormResize(Sender: TObject);
```

```
begin
```

```
Edit6.Text:=FloatToStr(StrToFloat(Edit5.Text)+Form1.Width);
```

```
Edit1.Text:=FloatToStr(StrToFloat(Edit3.Text)+Form1.Height);
```

```
Edit5.Position.X:=-35;
```

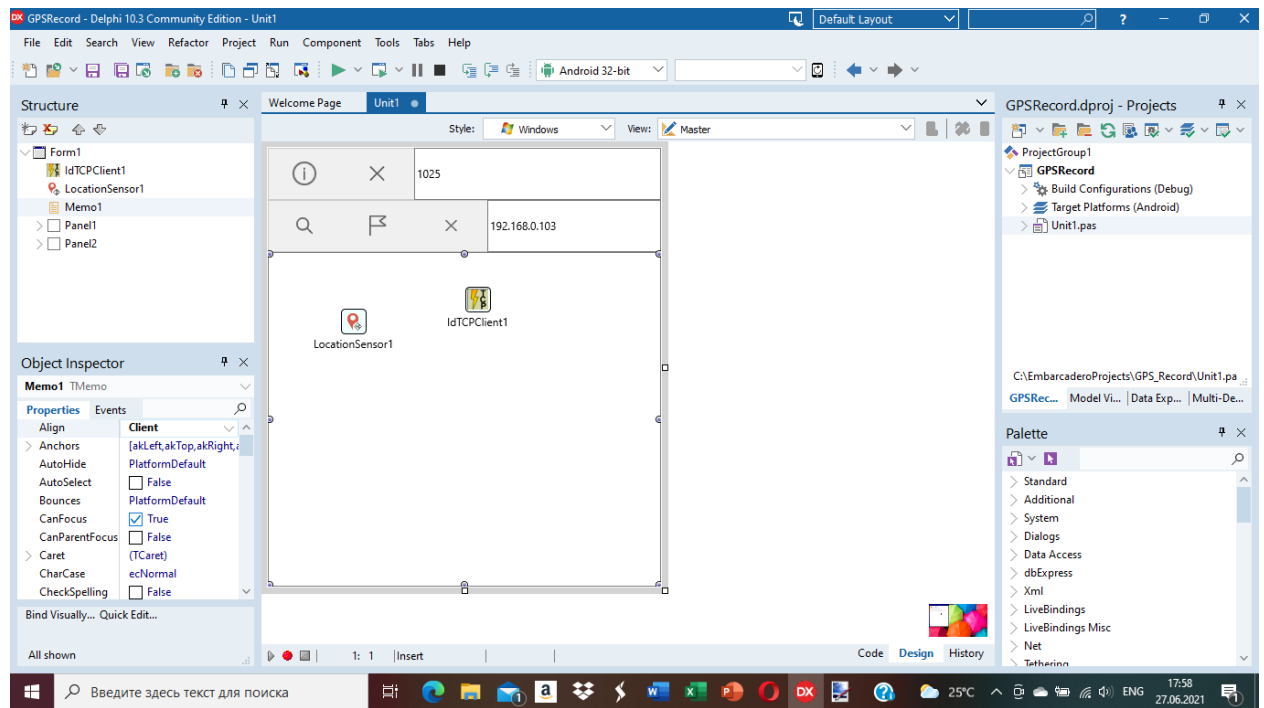
```
Edit5.Position.Y:=Rectangle1.Height/2;
```

```
Edit6.Position.X:=Rectangle1.Width-15;
```

```
Edit6.Position.Y:=Rectangle1.Height/2;
```

```
end;
```

1. Сбор данных



Собирает данные спутника и отправляет через локальную сеть по порту 1025.

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
begin
```

```
    LocationSensor1.Active := true;
```

```
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);
```

```
begin
```

```
    LocationSensor1.Active := false;
```

```
end;
```

```
procedure TForm1.Button3Click(Sender: TObject);
```

```
begin
```

```
    IdTCPClient1.Host := Edit1.Text;
```

```
    IdTCPClient1.Port := StrToInt(Edit2.Text);
```

```
    IdTCPClient1.Connect;
```

```
end;
```

```
procedure TForm1.Button4Click(Sender: TObject);
```

```
begin
```

```

IdTCPClient1.Socket.WriteLine(Memo1.Text);

end;

procedure TForm1.Button5Click(Sender: TObject);
begin
    IdTCPClient1.Disconnect;
end;

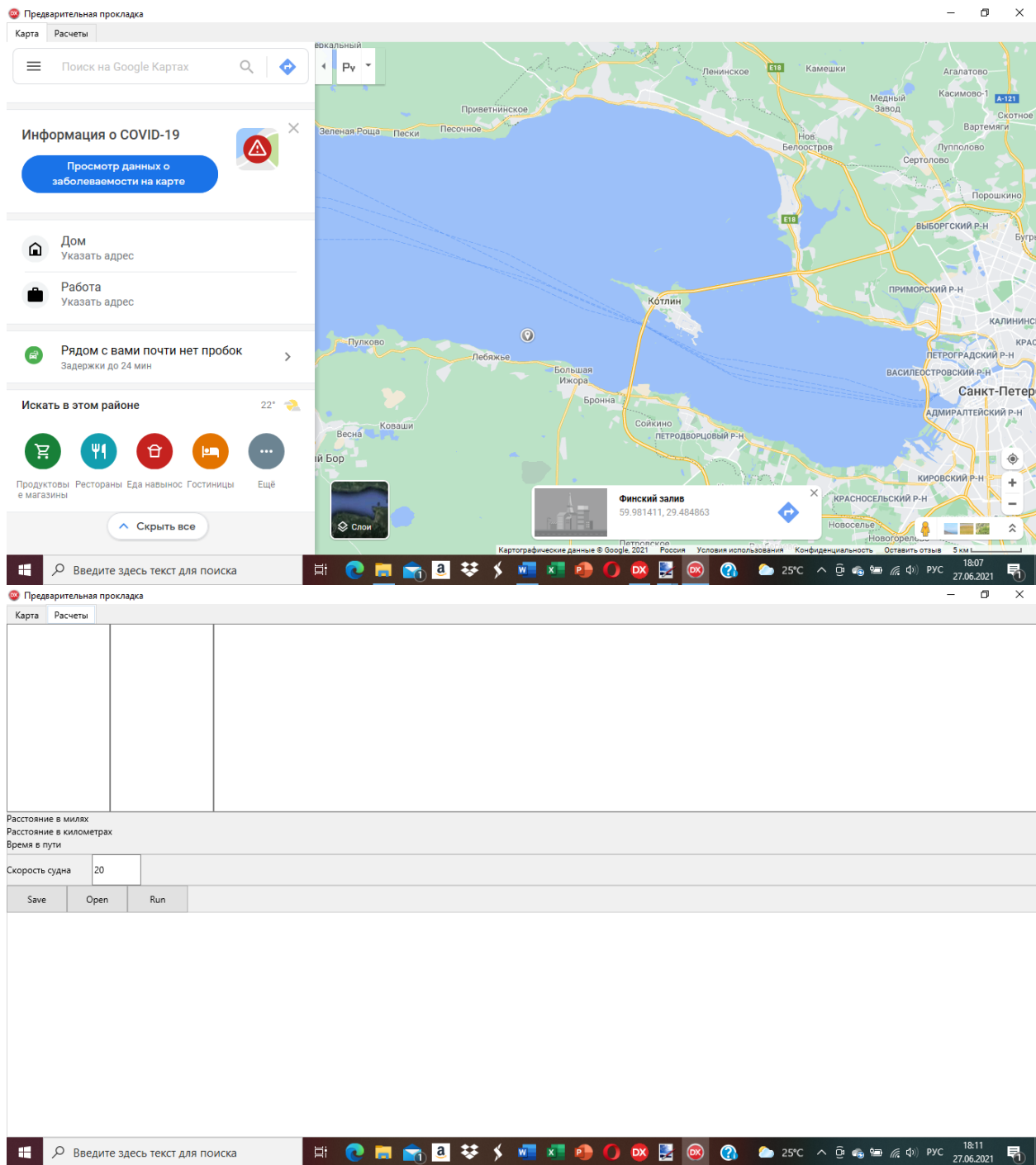
procedure TForm1.LocationSensor1LocationChanged(Sender: TObject);
    const OldLocation, NewLocation: TLocationCoord2D);
begin
    Memo1.Lines.Add(FloatToStr(LocationSensor1.Sensor.Latitude) + ' ' +
        FloatToStr(LocationSensor1.Sensor.Longitude) + ' ' +
        FloatToStrF(LocationSensor1.Sensor.Altitude, ffFixed, 5, 0) + ' ' +
        FloatToStrF(LocationSensor1.Sensor.TrueHeading, ffFixed, 3, 0) + ' ' +
        FloatToStrF(LocationSensor1.Sensor.Speed, ffFixed, 3, 0) + ' ' +
        FloatToStrF(LocationSensor1.Sensor.ErrorRadius, ffFixed, 5, 1));
end;

```

Собирается широта, долгота, высота, направление, скорость и ошибка места.

2. Предварительная прокладка.

Состоит из двух закладок. Первая карта Гугль. На ней можно отмечать точки. На второй закладке можно производить расчеты между точками. Курс и скорость и время в пути. Данные точек можно сохранять в текстовые файлы.



```

procedure TForm1.Button1Click(Sender: TObject);
begin
    if SaveDialog1.Execute then
        Memo1.Lines.SaveToFile(SaveDialog1.FileName);
    if SaveDialog2.Execute then
        Memo2.Lines.SaveToFile(SaveDialog2.FileName);
end;

```

```

procedure TForm1.Button2Click(Sender: TObject);
begin
    Memo1.Lines.Clear;
    Memo2.Lines.Clear;
    Memo3.Lines.Clear;

```

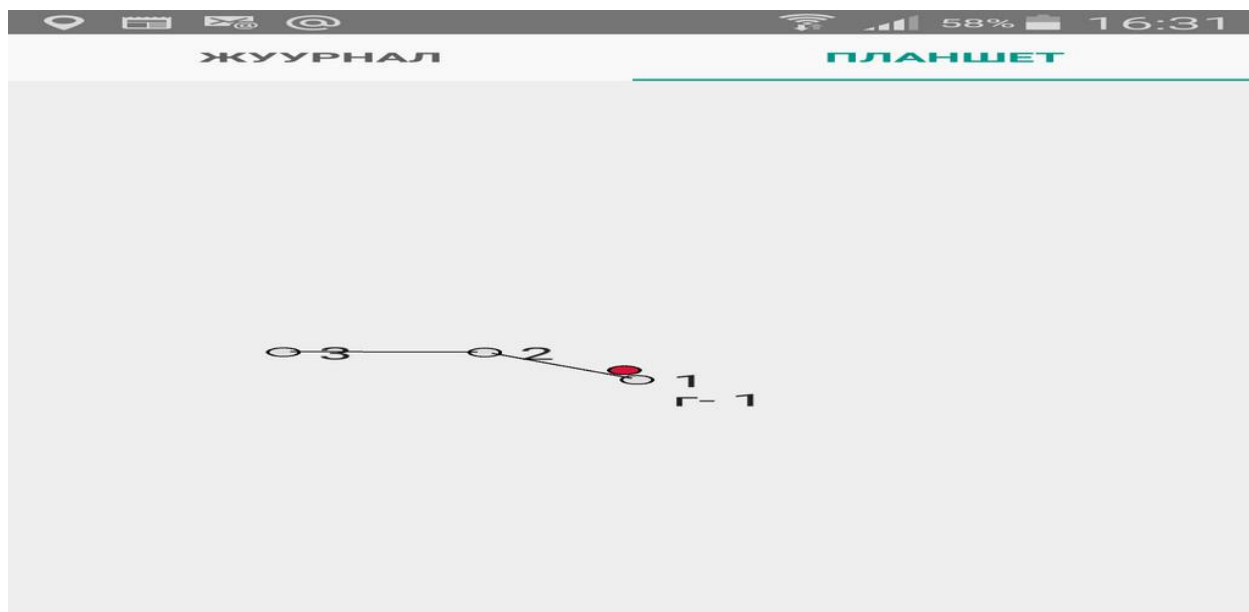
```

if OpenFileDialog1.Execute then
    Memo1.Lines.LoadFromFile(OpenDialog1.FileName);
if OpenFileDialog2.Execute then
    Memo2.Lines.LoadFromFile(OpenDialog2.FileName);
end;

procedure TForm1.Button3Click(Sender: TObject);
var
    i: integer;
    f, l, d, d1, K, f1, f2, l1, l2: extended;
begin
    Memo3.Lines.Clear;
    d := 0;
    for i := 0 to Memo1.Lines.Count - 2 do
        begin
            f1 := StrToFloat(Memo1.Lines[i]);
            f2 := StrToFloat(Memo1.Lines[i + 1]);
            l1 := StrToFloat(Memo2.Lines[i]);
            l2 := StrToFloat(Memo2.Lines[i + 1]);
            f := (f2 - f1) * 60;
            l := ((l2 - l1) * cos(DegToRad((f2 + f1) / 2))) * 60;
            d1 := Math.Hypot(f, l);
            K := RadToDeg(arctan(f / l));
            Memo3.Lines.Add('ПШ=' + FloatToStr(f) + ' РД=' + FloatToStr(l) + ' Курс=' +
                FloatToStrF(K, ffFixed, 5, 1) + ' Расстояние=' + FloatToStrF(d1,
                    ffFixed, 10, 0));
            d := d + d1;
        end;
    Label1.Text := FloatToStrF(d, ffFixed, 10, 0) + ' миль';
    Label4.Text := FloatToStrF(d*1.852, ffFixed, 10, 0) + ' км';
    Label3.Text := FloatToStrF(d / StrToFloat(Edit1.Text) / 24, ffFixed, 10, 1)
        + ' суток';
end;

```

3. Промер гидрографический.



ЖУУРНАЛ ПЛАНШЕТ

6627897,8 5 354927,4

Xs= 6626000 Xn= 6630000

Yw= 353000 Ye= 357000

галс 1 определ ение 4

место конец галса отпр... загруз...

1 1 59,73728 30,41903 6627801,1 354994,7
1 2 59,73728 30,41903 6627801,1 354994,7
1 3 59,73728 30,41903 6627801,1 354994,7

var

```

Form1: TForm1;

F1, L1, F2, L2, F_Old, L_Old: real;

B, L, H, dB, dL, dH, x1, y1, x2, y2, x, y, D, ugo1, x_old, y_old, ugo1_old,
    D_old, Heig: real;

p1, p2: TpointF;

EllipseRegion: TRectF;

```

```

const

```

```

    p = 206264.8062;

    ro: Extended = 206264.800023606351128218038600445;

```

```

implementation

```

```

{$R *.fmx}

```

```

procedure Preobr(Aa, Ab, Ea, Eb, B, Lat, H, dx, dy, dz, wx, wy, wz, mm: double);

```

```

var

```

```

    a, e2, da, de2, M, n: double;

```

```

begin

```

```

    a := (Ab + Aa) / 2;

```

```

    e2 := (sqr(Eb) + sqr(Ea)) / 2;

```

```

    da := Ab - Aa;

```

```

    de2 := sqr(Eb) - sqr(Ea);

```

```

    M := a * (1 - e2) * Power((1 - e2 * sqr(sin(B))), -3 / 2);

```

```

    n := a * Power((1 - e2 * sqr(sin(B))), -1 / 2);

```

```

    dB := p / (M + H) * ((n / a) * e2 * sin(B) * cos(B) * da +

```

```

        (sqr(n) / sqr(a) + 1) * n * sin(B) * cos(B) * (de2 / 2) -

```

```

        (dx * cos(L) + dy * sin(L)) * sin(B) + dz * cos(B)) - wx * sin(L) *

```

```

        (1 + e2 * cos(2 * B)) + wy * cos(L) * (1 + e2 * cos(2 * B)) - p * mm * e2 *

```

```

        sin(B) * cos(B);

```

```

    dL := (p / ((n + H) * cos(B))) * (-dx * sin(L) + dy * cos(L)) + tan(B) *

```

```

        (1 - e2) * (wx * cos(L) + wy * sin(L)) - wz;

```

```

dH := (-a / n) * da + n * sqr(sin(B)) * (de2 / 2) + (dx * cos(L) + dy * sin(L)
) * cos(B) + dz * sin(B) - n * e2 * sin(B) * cos(B) *
((wx / p) * sin(L) - (wy / p) * cos(L)) + (sqr(a) / n + H) * mm;
end;

// из БГС в СК
procedure TForm1.WGS_SK;
begin
    Lat := DegToRad(Lat);
    Lon := DegToRad(Lon);
    Preobr(6378137, 6378136, 1 / 298.257223563, 1 / 298.25784, Lat, Lon, Heig,
    1.08, 0.27, 0.9, 0, 0, 0.16, 0.12E-6); // из БГС в ПЗ
    dB := dB / 3600;
    dL := dL / 3600;
    Lat := RadToDeg(Lat) + dB;
    Lon := RadToDeg(Lon) + dL;
    Heig := Heig + dH;
    Lat := DegToRad(Lat);
    Lon := DegToRad(Lon);
    Preobr(6378136, 6378245, 1 / 298.25784, 1 / 298.3, Lat, Lon, Heig, -25.9,
    130.94, 81.76, 0, 0, 0, 0); // из ПЗ в СК 95
    dB := dB / 3600;
    dL := dL / 3600;
    Lat := RadToDeg(Lat) + dB;
    Lon := RadToDeg(Lon) + dL;
    Heig := Heig + dH;
    B := Lat;
    L := Lon;
end;

procedure TForm1.Button1Click(Sender: TObject);
var

```

```

E: TCircle;

Lab: TLabel;

Lin: Tline;

dx, dy, s, u, Xc, Yc: real;

begin
    // кружки
    E := TCircle.Create(Image1);
    E.Parent := Image1;
    Xc := Circle1.Position.x;
    Yc := Circle1.Position.y;
    E.Position.x := Xc - 5;
    E.Position.y := Yc - 5;
    E.Width := 10;
    E.Height := 10;
    ListBox1.Items.Add(FloatToStr(Xc));
    ListBox2.Items.Add(FloatToStr(Yc));
    // номер галса
    if Edit6.Text = '1' then
        begin
            Lab := TLabel.Create(Image1);
            Lab.Parent := Image1;
            Lab.Position.x := Xc + 10;
            Lab.Position.y := Yc + 10;
            Lab.Text := 'r- ' + Edit5.Text;
        end;
    // номер определения
    Lab := TLabel.Create(Image1);
    Lab.Parent := Image1;
    Lab.Position.x := Xc + 10;
    Lab.Position.y := Yc - 10;
    Lab.Text := Edit6.Text;
    // линия

```

```

if Edit6.Text <> '1' then
begin
    dx := Xc - ListBox1.Items[ListBox1.Count - 2].ToExtended;
    dy := Yc - ListBox2.Items[ListBox2.Count - 2].ToExtended;
    s := Math.Hypot(dx, dy);
    u := RadToDeg(arsin(dx / s));
    if -dy < 0 then
        u := 180 - u;
    if u < 0 then
        u := 360 + u;
    Lin := Tline.Create(Image1);
    Lin.Parent := Image1;
    Lin.Position.x := Xc;
    Lin.Position.y := Yc;
    Lin.RotationCenter.x := 0;
    Lin.RotationCenter.y := 0;
    Lin.RotationAngle := u;
    Lin.Width := 1;
    Lin.Height := s;
end;

Memo1.Lines.Add(Edit5.Text + ' ' + Edit6.Text + ' ' + FloatToStrF(F1, ffFixed,
    10, 5) + ' ' + FloatToStrF(L1, ffFixed, 10, 5) + ' ' + FloatToStrF(x,
    ffFixed, 10, 1) + ' ' + FloatToStrF(y, ffFixed, 10, 1));
Edit6.Text := intToStr(StrToInt(Edit6.Text) + 1);
{ WebBrowser1.URL := 'maps.google.com/maps?q=' + Label9.Text + ',' +
    Label10.Text + '&output=emded'; }
end;

```

```

procedure TForm1.Button2Click(Sender: TObject);

```

```

begin

```

```

    Edit6.Text := '1';

```

```

    Edit5.Text := intToStr(StrToInt(Edit5.Text) + 1);

```

```

ListBox1.Clear;

ListBox2.Clear;

end;


procedure TForm1.Button6Click(Sender: TObject);
begin
    IdTCPClient1.Host:=Edit7.Text;
    IdTCPClient1.Connect;
end;


procedure TForm1.Button7Click(Sender: TObject);
begin
    IdTCPClient1.Socket.WriteLine(Memo2.Text);
end;


procedure TForm1.FormShow(Sender: TObject);
begin
    FormatSettings.DecimalSeparator := '.';
end;


procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
    const OldLocation, NewLocation: TLocationCoord2D);
var
    n, Xn, Xs, Yw, Ye: real;
begin
    Label14.Text := 'Speed ' + FloatToStrF(LocationSensor1.Sensor.Speed,
        ffFixed, 4, 1);

    Label15.Text := 'Direct ' +
        FloatToStrF(LocationSensor1.Sensor.TrueHeading, ffFixed, 4, 0);

    Label18.Text := 'Height ' + FloatToStrF(LocationSensor1.Sensor.Altitude,
        ffFixed, 8, 1);

    F1 := NewLocation.Latitude;

```

```

L1 := NewLocation.Longitude;
Label9.Text := FloatToStr(F1);
Label10.Text := FloatToStr(L1);
Memo2.Lines.Add(Label9.Text + ' ' + Label10.Text + ' ' + Label1.Text + ' ' +
    Label2.Text + ' ' + Label14.Text + ' ' + Label15.Text + ' ' + Label18.Text);
Heig:= StrToFloat(Label18.Text);
n := int((int(L1) + 3) / 6);
WGS_SK(F1, L1, 0);
SK_Pr(B, L);
x1 := x;
y1 := y;
Label1.Text := FloatToStrF(x1, ffFixed, 10, 1);
Label2.Text := FloatToStr(n) + ' ' + FloatToStrF(y1, ffFixed, 10, 1);
Xs := StrToFloat(Edit1.Text);
Xn := StrToFloat(Edit2.Text);
Yw := StrToFloat(Edit3.Text);
Ye := StrToFloat(Edit4.Text);
Circle1.Position.y := (Image1.Height * (Xn - x1)) / (Xn - Xs);
Circle1.Position.x := (Image1.Width * (y1 - Yw)) / (Ye - Yw);
WebBrowser1.URL := 'https://www.google.com/search?client=opera&q=' +
    Label9.Text + '%2C' + Label10.Text + '&sourceid=opera&ie=UTF-8&oe=UTF-8';
end;

```

```

procedure TForm1.ShowShareSheetAction1BeforeExecute(Sender: TObject);
begin
    ShowShareSheetAction1.TextMessage := Label3.Text + Edit1.Text + ' ' +
        Label4.Text + Edit2.Text + ' ' + Label5.Text + Edit3.Text + ' ' +
        Label6.Text + Edit4.Text + ' ' + Memo1.Text;
end;

```

```

procedure TForm1.ShowShareSheetAction2BeforeExecute(Sender: TObject);
begin

```

```

ShowShareSheetAction2.TextMessage := Memo2.Text;

end;

// в прямоуго
procedure TForm1.SK_Pr;
var
    n, cos_sqr_B, a0, a4, a6, a3, a5, B_sek, N_Z, L0: Extended;
begin
    Lat := DegToRad(Lat); // Широта в радианах
    cos_sqr_B := sqr(cos(Lat));
    n := 6399698.902 - (21562.267 - (108.973 - 0.612 * cos_sqr_B) * cos_sqr_B) *
        cos_sqr_B;
    a0 := 32140.404 - (135.3302 - (0.7092 - 0.004 * cos_sqr_B) * cos_sqr_B) *
        cos_sqr_B;
    a4 := (0.25 + 0.00252 * cos_sqr_B) * cos_sqr_B - 0.04166;
    a6 := (0.166 * cos_sqr_B - 0.084) * cos_sqr_B;
    a3 := (0.3333333 + 0.001123 * cos_sqr_B) * cos_sqr_B - 0.1666667;
    a5 := 0.0083 - (0.1667 - (0.1968 + 0.004 * cos_sqr_B) * cos_sqr_B) *
        cos_sqr_B;
    N_Z := Round((Lon + 3) / 6);
    L0 := 6 * N_Z - 3;
    Lon := (Lon - L0) * 3600;
    Lon := Lon / ro;
    B_sek := RadToDeg(Lat) * 3600;
    x := 6367558.4969 * B_sek / ro - (a0 - (0.5 + (a4 + a6 * sqr(Lon)) * sqr(Lon))
        * sqr(Lon) * n) * sin(Lat) * cos(Lat);
    y := (1 + (a3 + a5 * sqr(Lon)) * sqr(Lon)) * Lon * n * cos(Lat);
    y := 500000 + y;
end;

procedure TForm1.TakePhotoFromLibraryAction1DidFinishTaking(Image: TBitmap);
begin

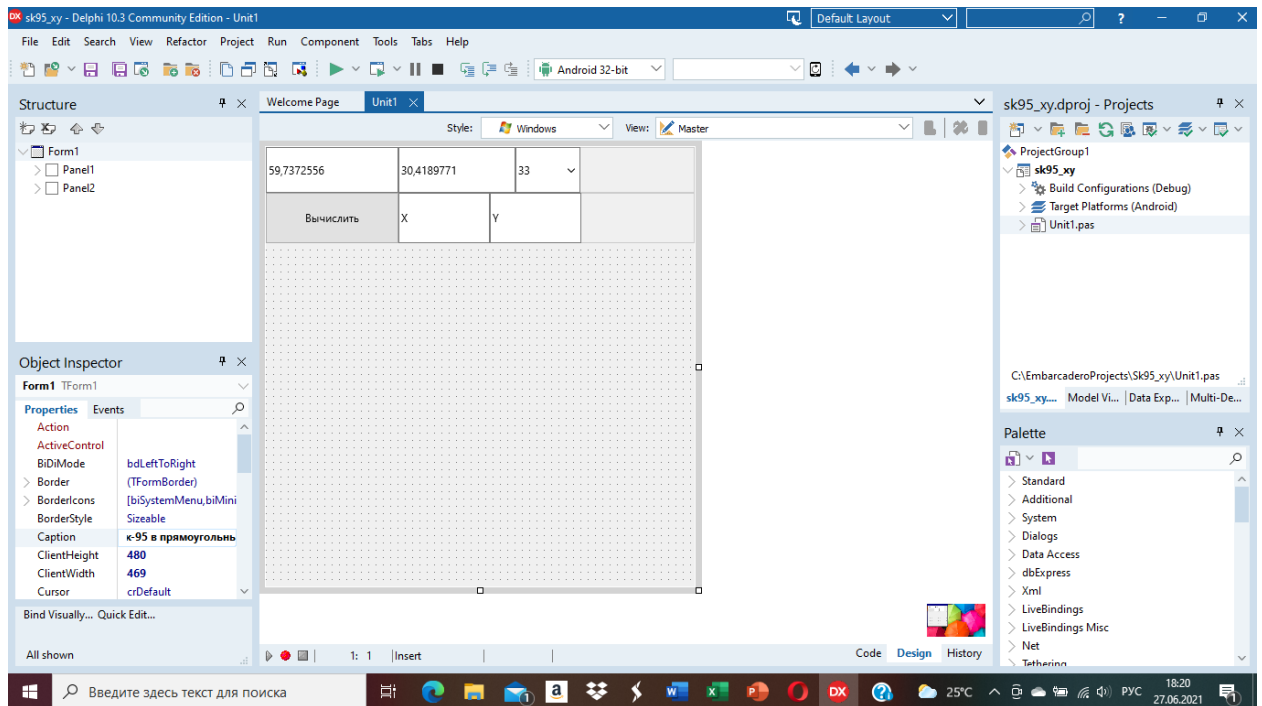
```

```
Image1.Bitmap.Assign(Image);
```

```
end;
```

Можно задать границы планшета и выполнять промер на мобильном телефоне. Данные отправлять по почте.

4. Перевод координат географических в прямоугольные.



```
const
```

```
ro: Extended = 206264.800023606351128218038600445;
```

```
implementation
```

```
{ $R *.fmx }
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var N,cos_sqr_B,a0,a4,a6,a3,a5,B_sek,x,y,N_Z,B,L:Extended;
```

```
begin
```

```
B:=StrToFloat(Edit1.Text);
```

```
B:=DegToRad(B);//Широта в радианах
```

```
cos_sqr_B:= sqr(cos(B));
```

```
N:=6399698.902-(21562.267-(108.973-0.612*cos_sqr_B)*cos_sqr_B)*cos_sqr_B;
```

```
a0:=32140.404-(135.3302-(0.7092-0.004*cos_sqr_B)*cos_sqr_B)*cos_sqr_B;
```

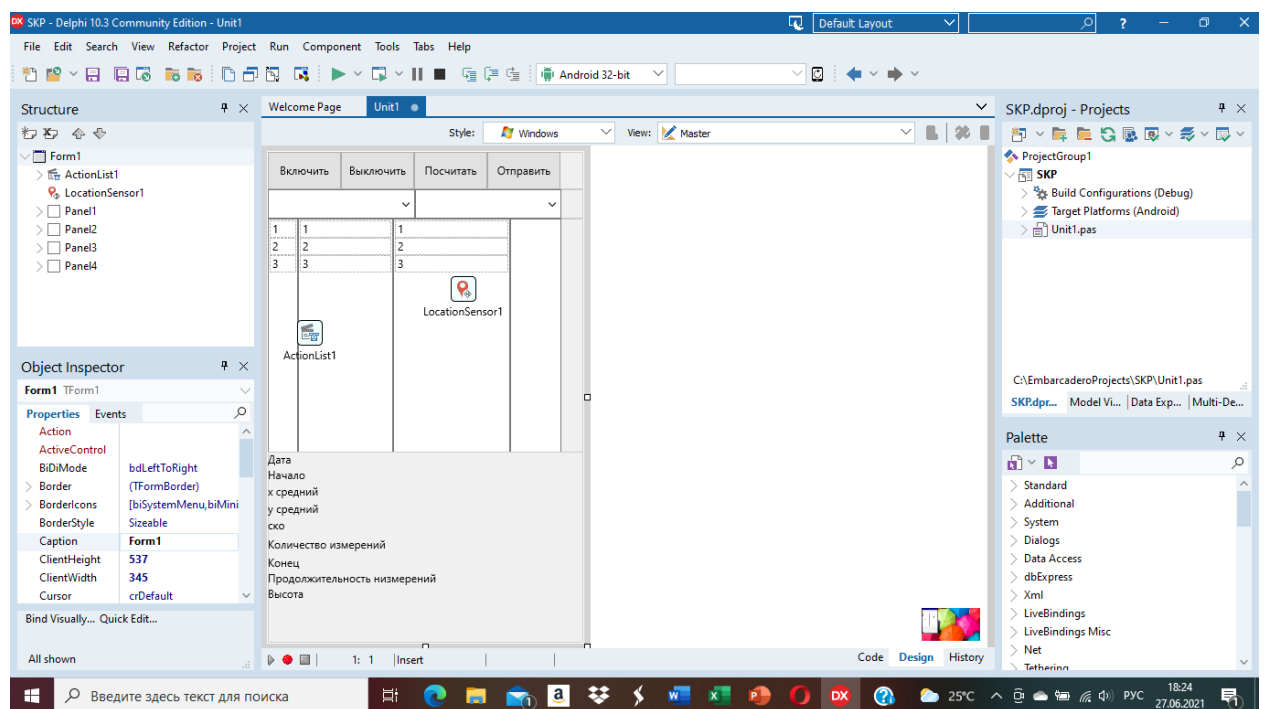
```
a4:=(0.25+0.00252*cos_sqr_B)*cos_sqr_B-0.04166;
```

```

a6:=(0.166*cos_sqr_B-0.084)*cos_sqr_B;
a3:=(0.3333333+0.001123*cos_sqr_B)*cos_sqr_B-0.1666667;
a5:=0.0083-(0.1667-(0.1968+0.004*cos_sqr_B)*cos_sqr_B)*cos_sqr_B;
L:=(StrToFloat(Edit2.text)-StrToFloat(ComboEdit1.text))*3600;
L:=L/ro;
B_sek:=StrToFloat(Edit1.text)*3600;
x:=6367558.4969*B_sek/ro-(a0-(0.5+(a4+a6*sqr(l))*sqr(l))*sqr(l)*N)*sin(B)*cos(B);
y:=(1+(a3+a5*sqr(l))*sqr(l))*l*N*cos(B);
y:=500000+y;
N_Z:=(StrToFloat(ComboEdit1.Text)+3)/6;
Edit4.Text:=FloatToStrF(x,ffFixed,10,2);
Edit5.Text:=FloatToStr(N_Z)+' '+FloatToStrF(y,ffFixed,11,2);
end;
end.

```

5. Вычисление ошибки спутниковых измерений.



var

Form1: TForm1;

F1, L1, F2, L2, F_Old, L_Old: real;

B, L, H, dB, dL, dH, x1, y1, x2, y2, x, y, D, ugol, x_old, y_old, ugol_old,

D_old: real;

n: integer;

const

p = 206264.8062;

ro: Extended = 206264.800023606351128218038600445;

implementation

{ \$R *.fmx }

// Вкл

procedure TForm1.Button1Click(Sender: TObject);

begin

LocationSensor1.Active := true;

n := 0;

ListBox1.Items.Clear;

ListBox2.Items.Clear;

ListBox3.Items.Clear;

Button1.Enabled := false;

Label5.Text := DateToStr(Date());

Label6.Text := TimeToStr(Time());

end;

// Выкл

procedure TForm1.Button2Click(Sender: TObject);

begin

LocationSensor1.Active := false;

Button1.Enabled := true;

Label7.Text := TimeToStr(Time());

end;

```

// Расчет
procedure TForm1.Button3Click(Sender: TObject);
var
  i, count: integer;
  x_sr, y_sr, x_sko, y_sko, sko, h_sr, h_sko: real;
begin
  count := ListBox1.Items.count - 1;

  x_sr := 0;
  y_sr := 0;
  x_sko := 0;
  y_sko := 0;
  h_sr:=0;
  h_sko:=0;
  Label1.Text := 'x средний';
  Label2.Text := 'y средний';
  Label3.Text := 'ско';
  Label4.Text := 'количество измерений';
  Label9.Text := 'высота';
  for i := 1 to ListBox1.Items.count - 1 do
  begin
    x_sr := x_sr + ListBox2.Items[i].ToExtended;
    y_sr := y_sr + ListBox3.Items[i].ToExtended;
    h_sr:=h_sr+LocationSensor1.Sensor.Altitude;
  end;
  x_sr := x_sr / count;
  y_sr := y_sr / count;
  h_sr:=h_sr/count;
  Label1.Text := Label1.Text + ' ' + FloatToStrF(x_sr,ffFixed,10,1);
  Label2.Text := Label2.Text + ' ' + FloatToStrF(y_sr,ffFixed,10,1);
  Label9.Text := Label9.Text + ' ' + FloatToStrF(h_sr,ffFixed,6,0);
  for i := 1 to ListBox1.Items.count - 1 do
  begin

```

```

x_sko := x_sko + sqr(ListBox2.Items[i].ToExtended - x_sr);
y_sko := y_sko + sqr(ListBox3.Items[i].ToExtended - y_sr);
h_sko:=h_sko+sqr(ListBox4.Items[i].ToExtended - h_sr);
end;

x_sko := sqrt(x_sko / (count - 1));
y_sko := sqrt(y_sko / (count - 1));
sko := sqrt(sqr(x_sko) + sqr(y_sko));
h_sko := sqrt(h_sko / (count - 1));

Label3.Text := Label3.Text + ' ' + FloatToStrF(sko,ffFixed,5,1)+' '+FloatToStrF(h_sko,ffFixed,6,0);
Label4.Text := Label4.Text + ' ' + intToStr(count - 1);
Label7.Text := TimeToStr(Time());
Label8.Text := TimeToStr(StrToTime(Label7.Text) - StrToTime(Label6.Text));
end;

```

```

procedure Preobr(Aa, Ab, Ea, Eb, B, Lat, H, dx, dy, dz, wx, wy, wz, mm: double);

```

```

var

```

```

    a, e2, da, de2, M, n: double;

```

```

begin

```

```

    a := (Ab + Aa) / 2;

```

```

    e2 := (sqr(Eb) + sqr(Ea)) / 2;

```

```

    da := Ab - Aa;

```

```

    de2 := sqr(Eb) - sqr(Ea);

```

```

    M := a * (1 - e2) * Power((1 - e2 * sqr(sin(B))), -3 / 2);

```

```

    n := a * Power((1 - e2 * sqr(sin(B))), -1 / 2);

```

```

    dB := p / (M + H) * ((n / a) * e2 * sin(B) * cos(B) * da +
        (sqr(n) / sqr(a) + 1) * n * sin(B) * cos(B) * (de2 / 2) -
        (dx * cos(L) + dy * sin(L)) * sin(B) + dz * cos(B)) - wx * sin(L) *
        (1 + e2 * cos(2 * B)) + wy * cos(L) * (1 + e2 * cos(2 * B)) - p * mm * e2 *
        sin(B) * cos(B);

```

```

    dL := (p / ((n + H) * cos(B))) * (-dx * sin(L) + dy * cos(L)) + tan(B) *
        (1 - e2) * (wx * cos(L) + wy * sin(L)) - wz;

```

```

    dH := (-a / n) * da + n * sqr(sin(B)) * (de2 / 2) + (dx * cos(L) + dy * sin(L)

```

```

) * cos(B) + dz * sin(B) - n * e2 * sin(B) * cos(B) *
((wx / p) * sin(L) - (wy / p) * cos(L)) + (sqr(a) / n + H) * mm;
end;

// из ВГС в СК
procedure TForm1.WGS_SK;
begin
    Lat := DegToRad(Lat);
    Lon := DegToRad(Lon);
    Preobr(6378137, 6378136, 1 / 298.257223563, 1 / 298.25784, Lat, Lon, Heig,
        1.08, 0.27, 0.9, 0, 0, 0.16, 0.12E-6); // из ВГС в ПЗ
    dB := dB / 3600;
    dL := dL / 3600;
    Lat := RadToDeg(Lat) + dB;
    Lon := RadToDeg(Lon) + dL;
    Heig := Heig + dH;
    Lat := DegToRad(Lat);
    Lon := DegToRad(Lon);
    Preobr(6378136, 6378245, 1 / 298.25784, 1 / 298.3, Lat, Lon, Heig, -25.9,
        130.94, 81.76, 0, 0, 0, 0); // из ПЗ в СК 95
    dB := dB / 3600;
    dL := dL / 3600;
    Lat := RadToDeg(Lat) + dB;
    Lon := RadToDeg(Lon) + dL;
    Heig := Heig + dH;
    B := Lat;
    L := Lon;
end;

procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
    const OldLocation, NewLocation: TLocationCoord2D);
var

```

```

X0: real;

begin
    n := n + 1;

    F1 := NewLocation.Latitude;

    L1 := NewLocation.Longitude;

    WGS_SK(F1, L1, 0);

    SK_Pr(B, L);

    x1 := x;

    y1 := y;

    ListBox1.Items.Add(intToStr(n));

    ListBox2.Items.Add(FloatToStrF(x1,ffFixed,10,1));

    ListBox3.Items.Add(FloatToStrF(y1,ffFixed,10,1));

    ListBox4.Items.Add(FloatToStrF(LocationSensor1.Sensor.Altitude,ffFixed,6,0));

end;

procedure TForm1.ShowShareSheetAction1BeforeExecute(Sender: TObject);
begin
    if (ComboEdit1.Text <> '') and (ComboEdit2.Text <> '') then
        ShowShareSheetAction1.TextMessage := 'Дата ' + Label5.Text + ', Начало ' +
            Label6.Text + ', марка телефона ' + ComboEdit1.Text +
            ', условия наблдений ' + ComboEdit2.Text + ', ско=' + Label3.Text +
            ', Xcp=' + Label1.Text + ', Ycp=' + Label2.Text + ' Высота=' + Label9.Text +
            ', Количество измерений ' + Label4.Text + ', Конец ' + Label7.Text +
            ', продолжительность ' + Label8.Text + ', Данные ' + ListBox2.Items.Text +
            '' + ListBox3.Items.Text;

end;

// в прямоуго
procedure TForm1.SK_Pr;
var
    n, cos_sqr_B, a0, a4, a6, a3, a5, B_sek, N_Z, L0: Extended;

begin

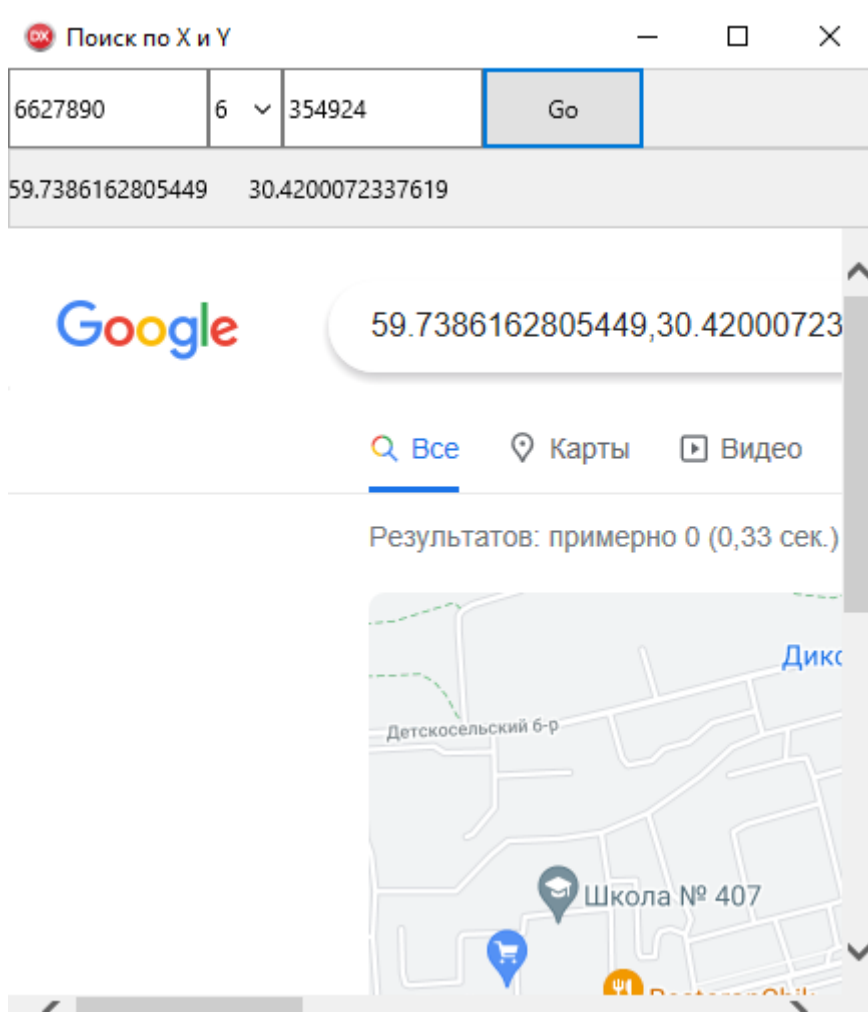
```

```

Lat := DegToRad(Lat); // Широта в радианах
cos_sqr_B := sqr(cos(Lat));
n := 6399698.902 - (21562.267 - (108.973 - 0.612 * cos_sqr_B) * cos_sqr_B) *
cos_sqr_B;
a0 := 32140.404 - (135.3302 - (0.7092 - 0.004 * cos_sqr_B) * cos_sqr_B) *
cos_sqr_B;
a4 := (0.25 + 0.00252 * cos_sqr_B) * cos_sqr_B - 0.04166;
a6 := (0.166 * cos_sqr_B - 0.084) * cos_sqr_B;
a3 := (0.3333333 + 0.001123 * cos_sqr_B) * cos_sqr_B - 0.1666667;
a5 := 0.0083 - (0.1667 - (0.1968 + 0.004 * cos_sqr_B) * cos_sqr_B) *
cos_sqr_B;
N_Z := Round((Lon + 3) / 6);
L0 := 6 * N_Z - 3;
Lon := (Lon - L0) * 3600;
Lon := Lon / ro;
B_sek := RadToDeg(Lat) * 3600;
x := 6367558.4969 * B_sek / ro - (a0 - (0.5 + (a4 + a6 * sqr(Lon)) * sqr(Lon))
* sqr(Lon) * n) * sin(Lat) * cos(Lat);
y := (1 + (a3 + a5 * sqr(Lon)) * sqr(Lon)) * Lon * n * cos(Lat);
y := 500000 + y;
end;
end.

```

6. Место на Гугле в прямоугольных координатах.



//из прямоугольных

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var n:integer;
```

```
B_grad,L0,y,beta,beta_grad,beta_rad,Bx,Bx_grad,sqr_cos_Bx,Nx,Bx_rad,b2,b3,b4,b5,z,l_mal,l_mal_grad:extended;
```

```
begin
```

```
y:=StrToFloat(Edit2.Text);
```

```
n:=StrToInt(ComboEdit1.Text);
```

```
L0:=6*n-3;
```

```
y:=y-500000;
```

```
beta:=(strToFloat(Edit1.text)/6367558.4969)*ro;
```

```
beta_grad:=beta/3600;
```

```
beta_rad:=DegToRad(beta_grad);
```

```
Bx:=beta+(50221746+(293622+(2350+22*sqr(cos(beta_rad)))*sqr(cos(beta_rad)))*sqr(cos(beta_rad)))*0.0000000001*sin(beta_rad)*cos(beta_rad)*ro;
```

```
Bx_grad:=Bx/3600;
```

```

Bx_rad:=DegToRad(Bx_grad);
sqr_cos_Bx:=sqr(cos(Bx_rad));
Nx:=6399698.902-(21562.267-(108.973-0.612*sqr_cos_Bx)*sqr_cos_Bx)*sqr_cos_Bx;
b2:=(0.5+0.003369*sqr_cos_Bx)*sin(Bx_rad)*cos(Bx_rad);
b3:=0.333333-(0.166667-0.001123*sqr_cos_Bx)*sqr_cos_Bx;
b4:=0.25+(0.16161+0.00562*sqr_cos_Bx)*sqr_cos_Bx;
b5:=0.2-(0.1667-0.0088*sqr_cos_Bx)*sqr_cos_Bx;
z:=y/(Nx*cos(Bx_rad));
l_mal:=(1-(b3-b5*sqr(z))*sqr(z))*z*ro;
l_mal_grad:=l_mal/3600;
L:=L0+l_mal_grad;
B:=Bx-(1-(b4-0.12*sqr(z))*sqr(z))*sqr(z)*b2*ro;
B_grad:=B/3600;
Label1.Text:=FloatToStr(B_grad);
Label1.Text:=StringReplace(Label1.Text,',',',',[rfReplaceAll, rfIgnoreCase]);
Label2.Text:=FloatToStr(L);
Label2.Text:=StringReplace(Label2.Text,',',',',[rfReplaceAll, rfIgnoreCase]);
//WebBrowser1.URL:='maps.google.com/maps?q='+Label1.Text+', '+Label2.Text+'&output=emded';
WebBrowser1.URL := 'https://www.google.com/search?client=opera&q=' +
Label1.Text + '%2C' + Label2.Text + '&sourceid=opera&ie=UTF-8&oe=UTF-8';
end;

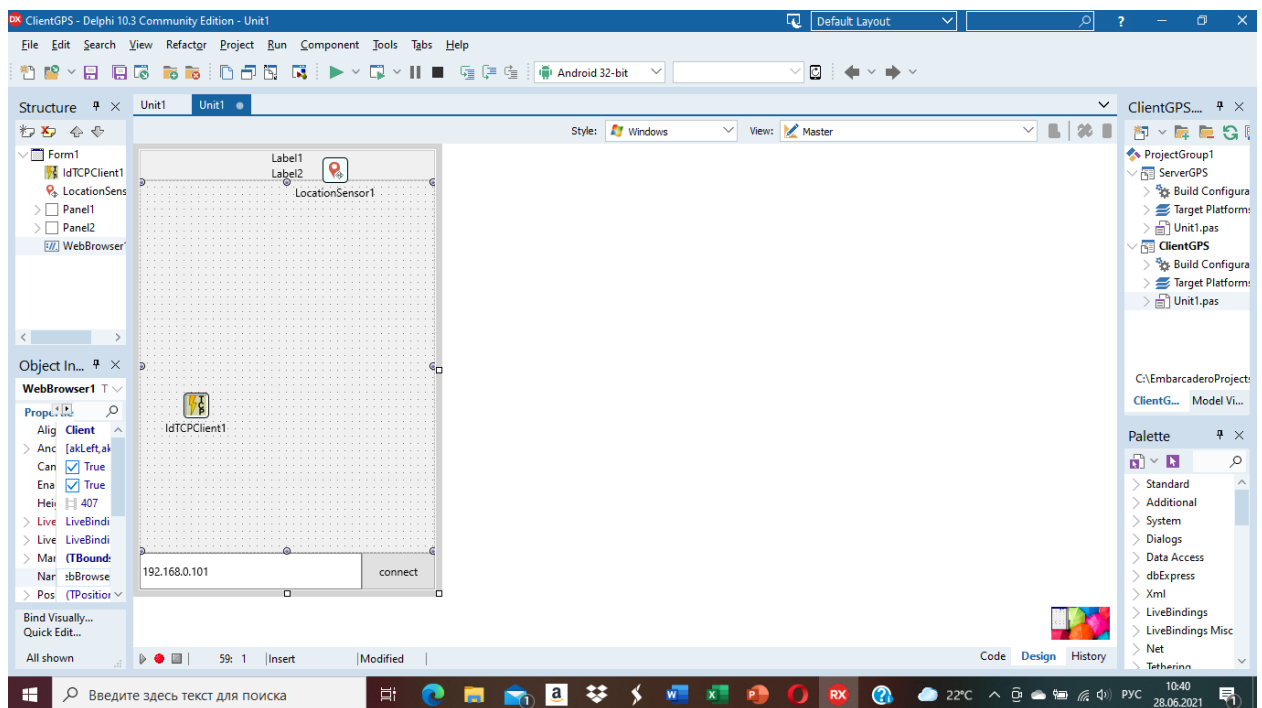
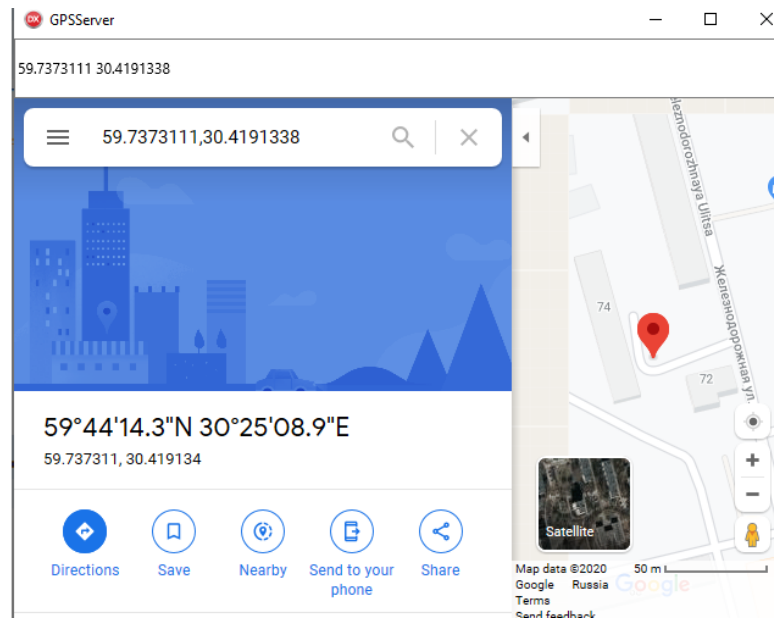
procedure TForm1.FormCreate(Sender: TObject);
begin
FormatSettings.DateSeparator:='.';
end;

```

7. Клиент сервер ГПС.

Программа передает место по спутнику на сервер и там отображается на гугль карте.

Клиент



```
//connect
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
begin
```

```
    IdTCPClient1.Host := Edit1.Text;
```

```
    IdTCPClient1.Connect;
```

```
    // отправка
```

```
    IdTCPClient1.Socket.WriteLine(f + ' ' + I);
```

```
end;
```

```
procedure TForm1.FormShow(Sender: TObject);
```

begin

```
FormatSettings.DecimalSeparator := '.';
```

end;

```
procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
```

```
const OldLocation, NewLocation: TLocationCoord2D);
```

begin

```
Label1.Text := NewLocation.Latitude.ToString;
```

```
Label2.Text := NewLocation.Longitude.ToString;
```

```
f:=Label1.Text;
```

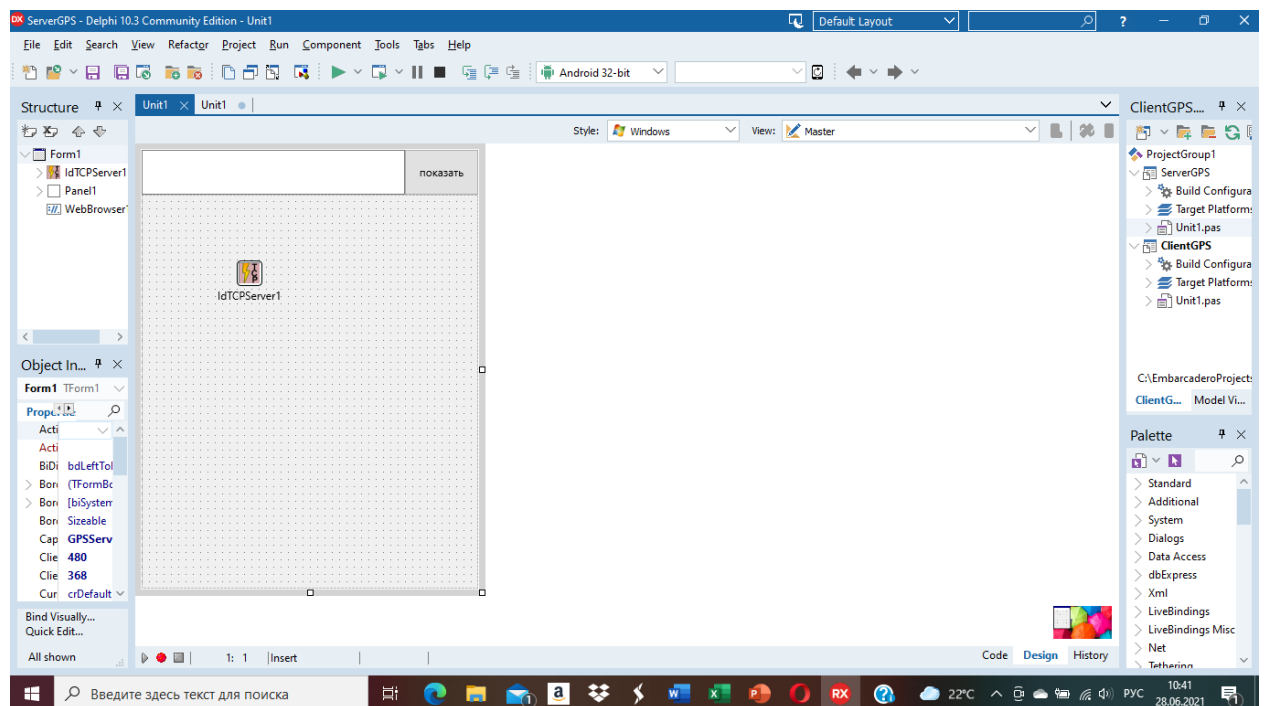
```
l:=Label2.Text;
```

```
WebBrowser1.URL := 'maps.google.com/maps?q=' + f + ',' + l
```

```
+ '&output=emded';
```

end;

Сербер:



s,f,l:string;

implementation

{ \$R *.fmx }

```

procedure TForm1.Button1Click(Sender: TObject);
begin
    WebBrowser1.URL := 'maps.google.com/maps?q=' + f + ',' + l
        + '&output=emded';
end;

```

```

procedure TForm1.FormDestroy(Sender: TObject);
begin
    IdTCPServer1.Active := false;
end;

```

```

procedure TForm1.FormShow(Sender: TObject);
begin
    IdTCPServer1.Active := true;
end;

```

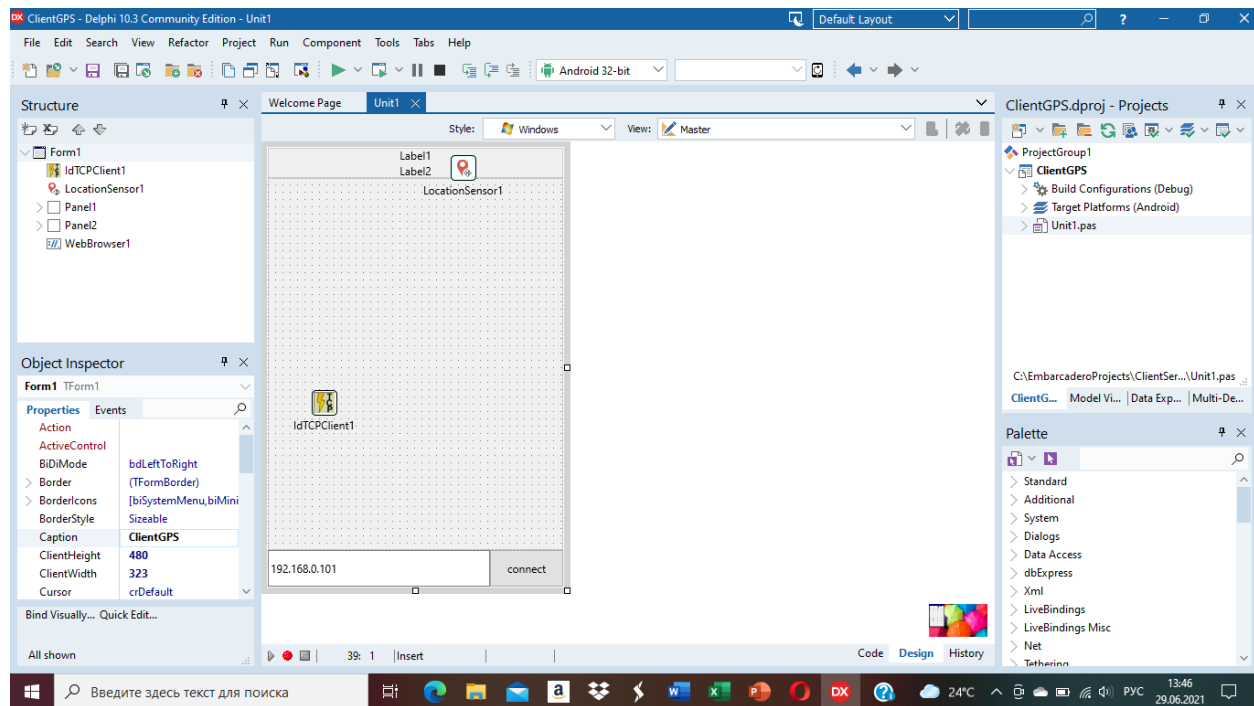
```

procedure TForm1.IdTCPServer1Execute(AContext: TIdContext);
begin
    Edit1.Text := AContext.Connection.Socket.ReadLn;
    s := Edit1.Text;
    f := Copy(s, 1, pos(' ', s) - 1);
    delete(s, 1, pos(' ', s));
    l := Copy(s, 1, Length(s));
end;

```

Есть еще программа, передающая поправки к координатам на известной точке. Есть вариант сервер на ровере и есть клиент на ровере.

- 8. Исправление координат поправками полученными на твердой точке.**
Клиент программы.



var

```
Form1: TForm1; f,l,f1,f2,l1,l2:string;
```

implementation

```
{ $R *.fmx }
```

```
//connect
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
begin
```

```
  IdTCPClient1.Host := Edit1.Text;
```

```
  IdTCPClient1.Connect;
```

```
  // отправка
```

```
  IdTCPClient1.Socket.WriteLine(f + ' ' + l);
```

```
end;
```

```
procedure TForm1.FormShow(Sender: TObject);
```

```
begin
```

```
  FormatSettings.DecimalSeparator := '.';
```

```
end;
```

```
procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
```

```
  const OldLocation, NewLocation: TLocationCoord2D);
```

```
begin
```

```
  Label1.Text := NewLocation.Latitude.ToString;
```

```
  Label2.Text := NewLocation.Longitude.ToString;
```

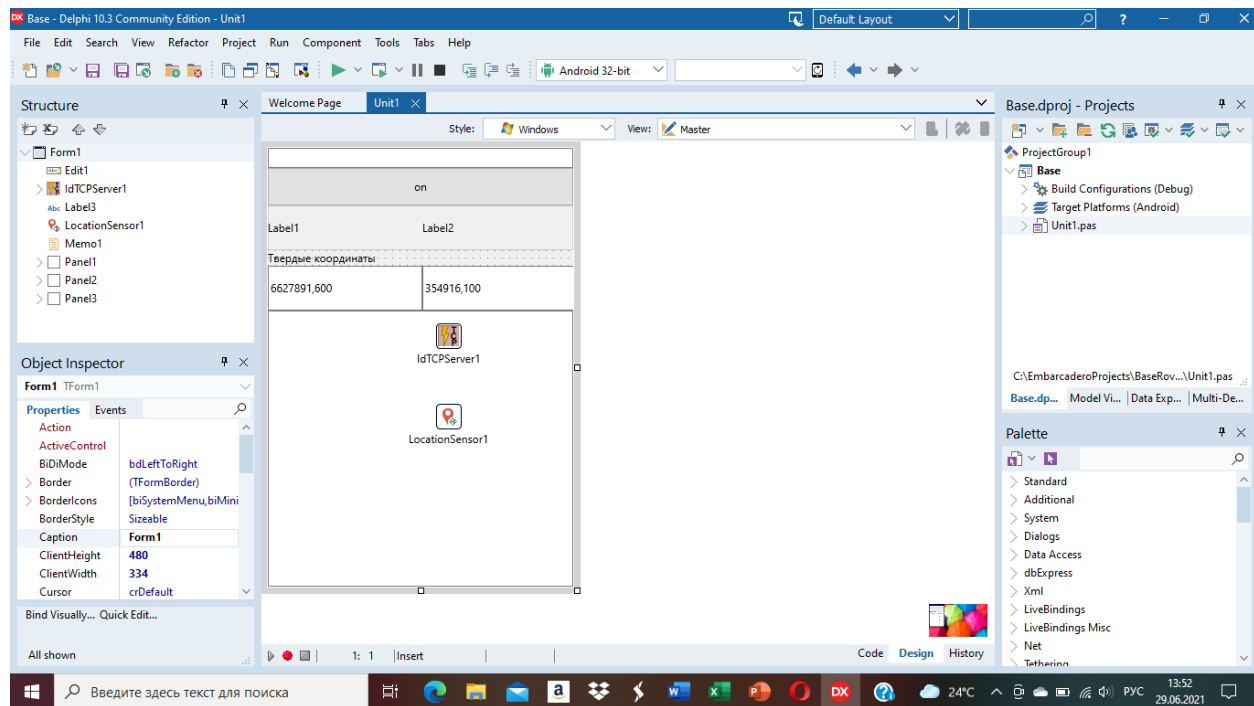
```
  f:=Label1.Text;
```

```
  l:=Label2.Text;
```

```
  WebBrowser1.URL := 'maps.google.com/maps?q=' + f + ' ' + l  
    + '&output=emded';
```

```
end;
```

Программа сервер.



var

Form1: TForm1;

B, L, H, dB, dL, dH, x1, y1, x2, y2, x, y, D, ugod, x_old, y_old, ugod_old, D_old: real;

F1, L1, F2, L2, F_Old, L_Old: real;

dX, dY: real;

const

p = 206264.8062;

ro: Extended = 206264.800023606351128218038600445;

implementation

{ \$R *.fmx }

procedure Preobr(Aa, Ab, Ea, Eb, B, Lat, H, dX, dY, dz, wx, wy, wz, mm: double);

var

a, e2, da, de2, M, n: double;

begin

a := (Ab + Aa) / 2;

e2 := (sqr(Eb) + sqr(Ea)) / 2;

da := Ab - Aa;

de2 := sqr(Eb) - sqr(Ea);

M := a * (1 - e2) * Power((1 - e2 * sqr(sin(B))), -3 / 2);

n := a * Power((1 - e2 * sqr(sin(B))), -1 / 2);

dB := p / (M + H) * ((n / a) * e2 * sin(B) * cos(B) * da +

(sqr(n) / sqr(a) + 1) * n * sin(B) * cos(B) * (de2 / 2) -

(dX * cos(L) + dY * sin(L)) * sin(B) + dz * cos(B)) - wx * sin(L) *

(1 + e2 * cos(2 * B)) + wy * cos(L) * (1 + e2 * cos(2 * B)) - p * mm * e2 *

sin(B) * cos(B);

dL := (p / ((n + H) * cos(B))) * (-dX * sin(L) + dY * cos(L)) + tan(B) *

(1 - e2) * (wx * cos(L) + wy * sin(L)) - wz;

```

dH := (-a / n) * da + n * sqr(sin(B)) * (de2 / 2) + (dX * cos(L) + dY * sin(L)
) * cos(B) + dz * sin(B) - n * e2 * sin(B) * cos(B) *
((wx / p) * sin(L) - (wy / p) * cos(L)) + (sqr(a) / n + H) * mm;
end;

procedure TForm1.SK_Pr;
var
  n, cos_sqr_B, a0, a4, a6, a3, a5, B_sek, N_Z, L0: Extended;
begin
  Lat := DegToRad(Lat); // Широта в радианах
  cos_sqr_B := sqr(cos(Lat));
  n := 6399698.902 - (21562.267 - (108.973 - 0.612 * cos_sqr_B) * cos_sqr_B) *
    cos_sqr_B;
  a0 := 32140.404 - (135.3302 - (0.7092 - 0.004 * cos_sqr_B) * cos_sqr_B) *
    cos_sqr_B;
  a4 := (0.25 + 0.00252 * cos_sqr_B) * cos_sqr_B - 0.04166;
  a6 := (0.166 * cos_sqr_B - 0.084) * cos_sqr_B;
  a3 := (0.3333333 + 0.001123 * cos_sqr_B) * cos_sqr_B - 0.1666667;
  a5 := 0.0083 - (0.1667 - (0.1968 + 0.004 * cos_sqr_B) * cos_sqr_B) *
    cos_sqr_B;
  N_Z := Round((Lon + 3) / 6);
  L0 := 6 * N_Z - 3;
  Lon := (Lon - L0) * 3600;
  Lon := Lon / ro;
  B_sek := RadToDeg(Lat) * 3600;
  x := 6367558.4969 * B_sek / ro - (a0 - (0.5 + (a4 + a6 * sqr(Lon)) * sqr(Lon))
    * sqr(Lon) * n) * sin(Lat) * cos(Lat);
  y := (1 + (a3 + a5 * sqr(Lon)) * sqr(Lon)) * Lon * n * cos(Lat);
  y := 500000 + y;
end;

// из ВГС в СК
procedure TForm1.WGS_SK;
begin
  Lat := DegToRad(Lat);
  Lon := DegToRad(Lon);
  Preobr(6378137, 6378136, 1 / 298.257223563, 1 / 298.25784, Lat, Lon, Heig,
    1.08, 0.27, 0.9, 0, 0, 0.16, 0.12E-6); // из ВГС в ПЗ
  dB := dB / 3600;
  dL := dL / 3600;
  Lat := RadToDeg(Lat) + dB;
  Lon := RadToDeg(Lon) + dL;
  Heig := Heig + dH;
  Lat := DegToRad(Lat);
  Lon := DegToRad(Lon);
  Preobr(6378136, 6378245, 1 / 298.25784, 1 / 298.3, Lat, Lon, Heig, -25.9,
    130.94, 81.76, 0, 0, 0, 0); // из ПЗ в СК 95
  dB := dB / 3600;
  dL := dL / 3600;
  Lat := RadToDeg(Lat) + dB;

```

```

Lon := RadToDeg(Lon) + dL;
Heig := Heig + dH;
B := Lat;
L := Lon;
end;

```

```

procedure TForm1.Button1Click(Sender: TObject);
begin
  LocationSensor1.Active := true;
  IdTCPServer1.Active := true;
end;

```

```

procedure TForm1.IdTCPServer1Execute(AContext: TIdContext);
var
  s, _x, _y: string;
  Xispr, Yispr: real;
begin
  Edit1.Text := AContext.Connection.Socket.ReadLn;
  s := Edit1.Text;
  _x := Copy(s, 1, pos(' ', s) - 1);
  delete(s, 1, pos(' ', s));
  _y := Copy(s, 1, Length(s));
  Xispr := StrToFloat(_x) + dX;
  Yispr := StrToFloat(_y) + dY;
  AContext.Connection.Socket.WriteLn(FloatToStrF(Xispr, ffFixed, 11, 3) + ' ' +
    FloatToStrF(Yispr, ffFixed, 11, 3));
end;

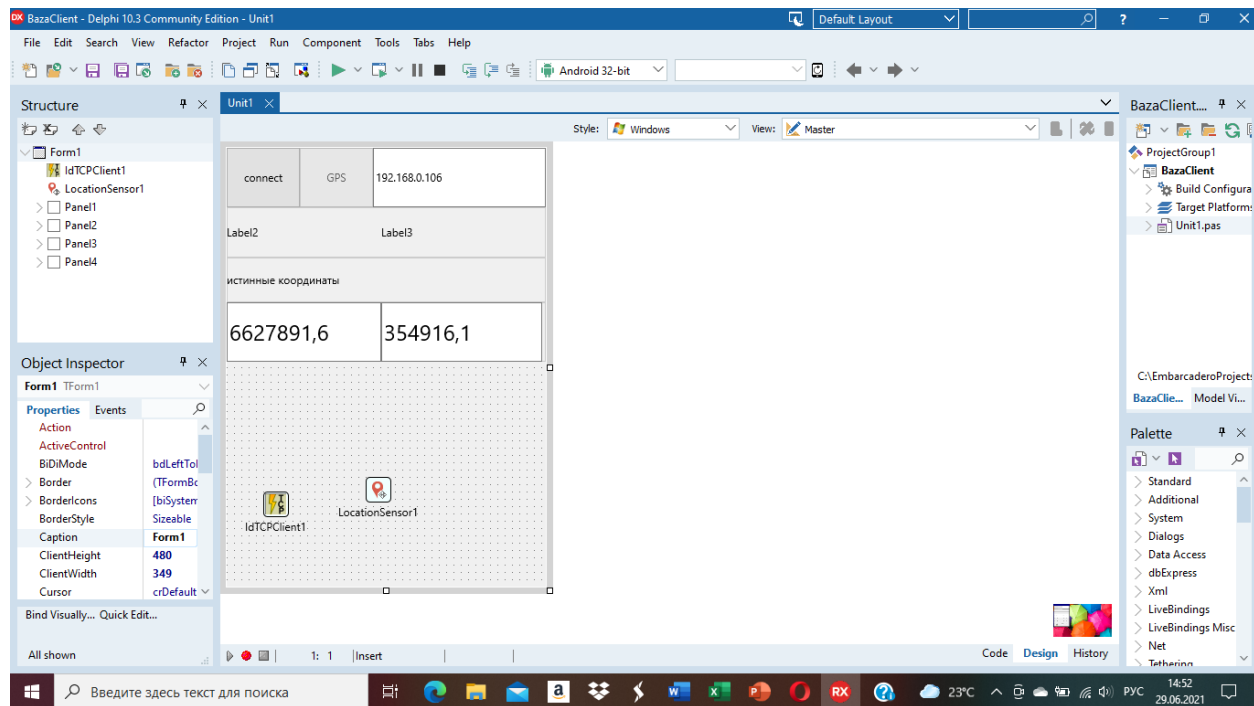
```

```

procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
  const OldLocation, NewLocation: TLocationCoord2D);
begin
  F1 := NewLocation.Latitude;
  L1 := NewLocation.Longitude;
  WGS_SK(F1, L1, 0);
  SK_Pr(B, L);
  x1 := x;
  y1 := y;
  Label1.Text := FloatToStr(x1);
  Label2.Text := FloatToStr(y1);
  dX := StrToFloat(Edit2.Text) - x1;
  dY := StrToFloat(Edit3.Text) - y1;
  Memo1.Lines.Add(FloatToStrF(dX, ffFixed, 11, 3) + ' ' + FloatToStrF(dY,
    ffFixed, 11, 3));
end;
end.

```

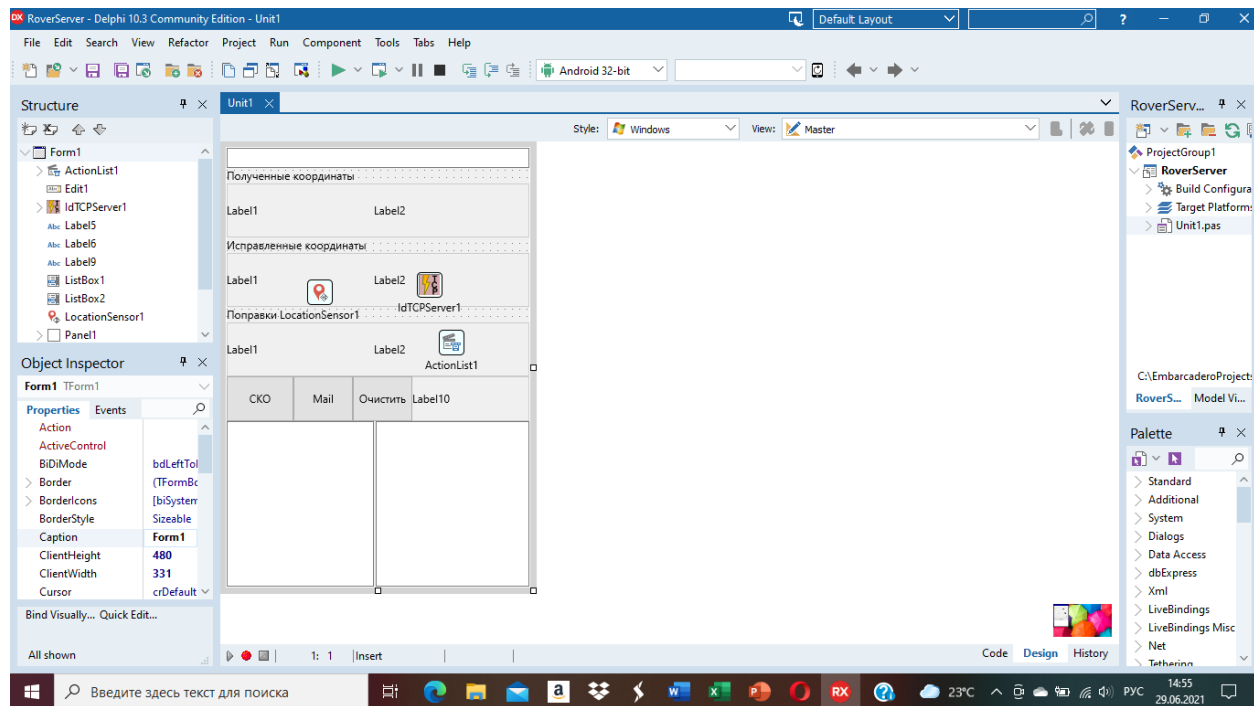
- 9. Исправление координат (еще одна. Клиент на базе)**
 Программа база.



```
procedure TForm1.Button2Click(Sender: TObject);
begin
  LocationSensor1.Active:=true;
end;
```

```
procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
  const OldLocation, NewLocation: TLocationCoord2D);
var dX,dY:real;
begin
  F1 := NewLocation.Latitude;
  L1 := NewLocation.Longitude;
  WGS_SK(F1, L1, 0);
  SK_Pr(B, L);
  x1 := x;
  y1 := y;
  Label2.Text:=FloatToStr(x1);
  Label3.Text:=FloatToStr(y1);
  dX:=StrToFloat(Edit1.Text)-x1;
  dY:=StrToFloat(Edit2.Text)-y1;
  IdTCPClient1.Socket.WriteLine(FloatToStr(dX)+' '+FloatToStr(dY));
end;
```

Исправленные координаты спутника отправляет клиенту. Ровер является сервером



```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
```

```
    i, count: integer;
```

```
    x_sr, y_sr, x_sko, y_sko, sko: real;
```

```
begin
```

```
    count := ListBox1.Items.Count;
```

```
    x_sr := 0;
```

```
    y_sr := 0;
```

```
    x_sko := 0;
```

```
    y_sko := 0;
```

```
    Label10.Text := 'ско';
```

```
    for i := 0 to ListBox1.Items.Count - 1 do
```

```
    begin
```

```
        x_sr := x_sr + ListBox1.Items[i].ToExtended;
```

```
        y_sr := y_sr + ListBox2.Items[i].ToExtended;
```

```
    end;
```

```
    x_sr := x_sr / count;
```

```
    y_sr := y_sr / count;
```

```
    for i := 1 to ListBox1.Items.Count - 1 do
```

```
    begin
```

```
        x_sko := x_sko + sqrt(ListBox1.Items[i].ToExtended - x_sr);
```

```
        y_sko := y_sko + sqrt(ListBox2.Items[i].ToExtended - y_sr);
```

```
    end;
```

```
    x_sko := sqrt(x_sko / count);
```

```
    y_sko := sqrt(y_sko / count);
```

```
    sko := sqrt(sqrt(x_sko) + sqrt(y_sko));
```

```
    Label10.Text := Label10.Text + ' ' + FloatToStrF(sko, ffFixed, 10, 3);
```

```
end;
```

```
procedure TForm1.Button3Click(Sender: TObject);
```

```
begin
```

```
    ListBox1.Items.Clear;
```

```

ListBox2.Items.Clear;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
  IdTCPServer1.Active := true;
end;

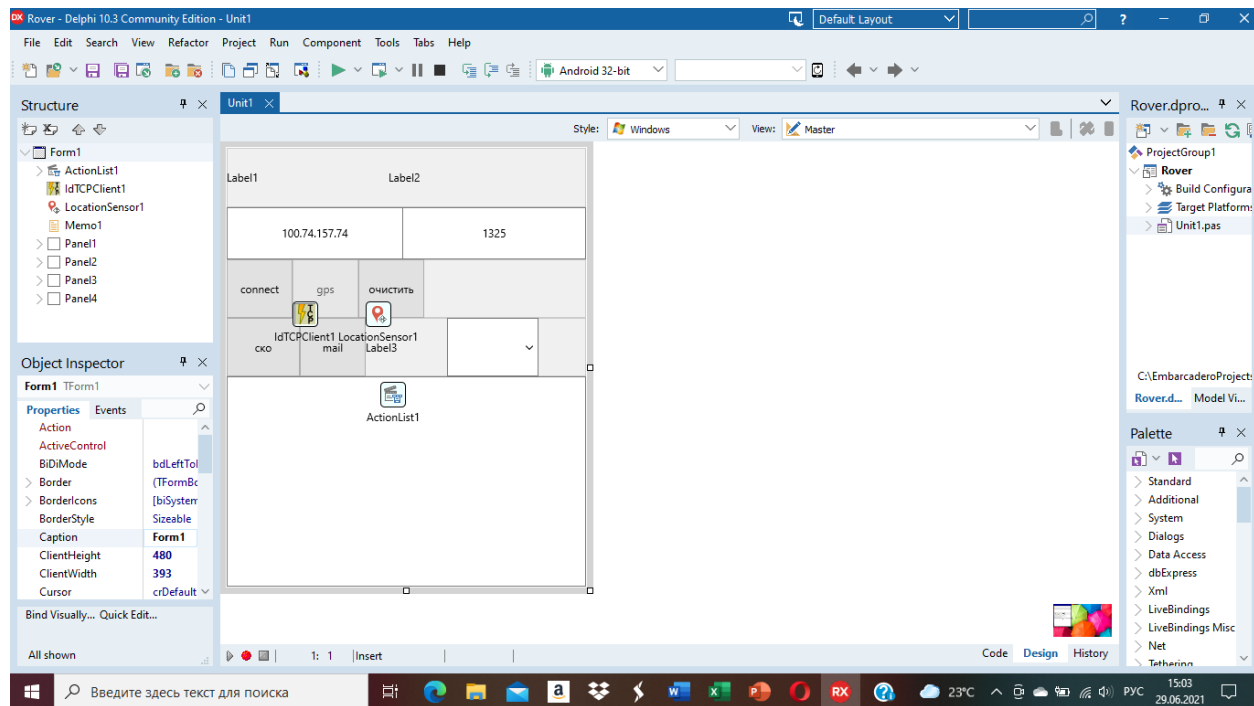
procedure TForm1.IdTCPServer1Execute(AContext: TIdContext);
var
  s: string;
begin
  Edit1.Text := AContext.Connection.Socket.ReadLn;
  s := Edit1.Text;
  _dx := Copy(s, 1, pos(' ', s) - 1);
  delete(s, 1, pos(' ', s));
  _dy := Copy(s, 1, Length(s));
  Label7.Text := _dx;
  Label8.Text := _dy;
  dX := StrToFloat(_dx);
  dY := StrToFloat(_dy);
end;

procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
  const OldLocation, NewLocation: TLocationCoord2D);
begin
  F1 := NewLocation.Latitude;
  L1 := NewLocation.Longitude;
  WGS_SK(F1, L1, 0);
  SK_Pr(B, L);
  x1 := x;
  y1 := y;
  Label1.Text := FloatToStr(x1);
  Label2.Text := FloatToStr(y1);
  Label3.Text := FloatToStr(x1 + dX);
  Label4.Text := FloatToStr(y1 + dY);
  ListBox1.Items.Add(Label3.Text);
  ListBox2.Items.Add(Label4.Text);
end;

```

10. Еще вариант базы и ровера.

Ровер отправляет на базу



```

procedure TForm1.Button1Click(Sender: TObject);
begin
  IdTCPClient1.Host := Edit1.Text;
  IdTCPClient1.Port:=StrToInt(Edit2.Text);
  IdTCPClient1.Connect;
  Button2.Enabled := true;
end;

```

```

procedure TForm1.Button2Click(Sender: TObject);
begin
  LocationSensor1.Active := true;
end;

```

```

procedure TForm1.Button4Click(Sender: TObject);
var
  i: Integer;
  sredneeX, sredneeY, skoX, skoY, sko: real;
  ss, xx, yy: string;
  count: Integer;
begin
  sredneeX := 0;
  sredneeY := 0;
  skoX := 0;
  skoY := 0;
  count := Memo1.Lines.count;
  Button4.Text := intToStr(count);
  for i := 0 to Memo1.Lines.count - 1 do
  begin
    ss := Memo1.Lines[i];
    xx := Copy(ss, 1, pos(' ', ss) - 1);
    Trim(xx);
    delete(ss, 1, pos(' ', ss));

```

```

yy := Copy(ss, 1, length(ss));
Trim(yy);
sredneeX := sredneeX + StrToFloat(xx);
sredneeY := sredneeY + StrToFloat(yy);
end;
sredneeX := sredneeX / count;
sredneeY := sredneeY / count;
for i := 0 to Memo1.Lines.count - 1 do
begin
ss := Memo1.Lines[i];
xx := Copy(ss, 1, pos(' ', ss) - 1);
Trim(xx);
delete(ss, 1, pos(' ', ss));
yy := Copy(ss, 1, length(ss));
Trim(yy);
skoX := skoX + sqr(StrToFloat(xx) - sredneeX);
skoY := skoY + sqr(StrToFloat(yy) - sredneeY);
end;
skoX := sqrt(skoX / count);
skoY := sqrt(skoY / count);
sko := sqrt(sqr(skoX) + sqr(skoY));
Label3.Text := FloatToStrF(sko, ffFixed, 10, 3);
end;

```

```

procedure TForm1.Button5Click(Sender: TObject);
begin
Memo1.Lines.Clear;
end;

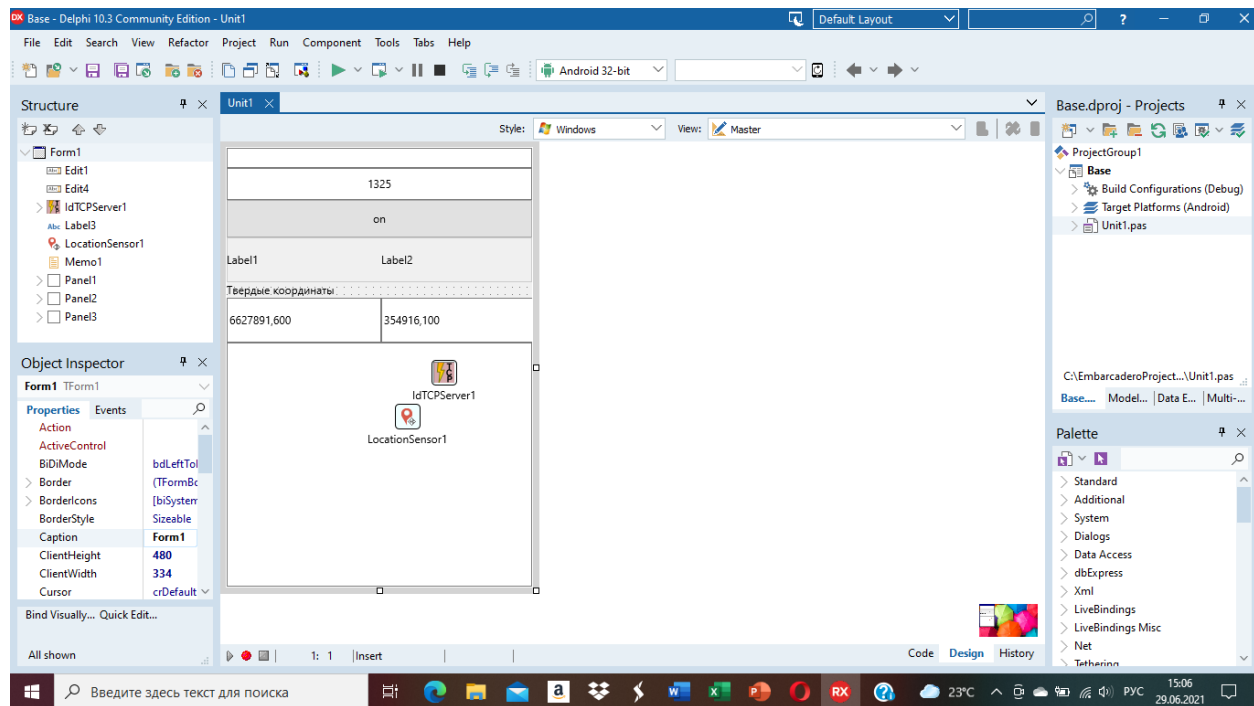
```

```

procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
const OldLocation, NewLocation: TLocationCoord2D);
begin
F1 := NewLocation.Latitude;
L1 := NewLocation.Longitude;
WGS_SK(F1, L1, 0);
SK_Pr(B, L);
x1 := x;
y1 := y;
Label1.Text := FloatToStr(x1);
Label2.Text := FloatToStr(y1);
IdTCPClient1.Socket.WriteLine(Label1.Text + ' ' + Label2.Text);
Memo1.Lines.Add(IdTCPClient1.Socket.ReadLn);
end;

```

На базе твердые координаты.



```

procedure TForm1.Button1Click(Sender: TObject);
begin
  LocationSensor1.Active := true;
  IdTCPServer1.DefaultPort:=StrToInt(Edit4.Text);
  IdTCPServer1.Active := true;
end;

```

```

procedure TForm1.IdTCPServer1Execute(AContext: TIdContext);
var
  s, _x, _y: string;
  Xispr, Yispr: real;
begin
  Edit1.Text := AContext.Connection.Socket.ReadLn;
  s := Edit1.Text;
  _x := Copy(s, 1, pos(' ', s) - 1);
  delete(s, 1, pos(' ', s));
  _y := Copy(s, 1, Length(s));
  Xispr := StrToFloat(_x) + dX;
  Yispr := StrToFloat(_y) + dY;
  AContext.Connection.Socket.WriteLn(FloatToStrF(Xispr, ffFixed, 11, 3) + ' ' +
    FloatToStrF(Yispr, ffFixed, 11, 3));
end;

```

```

procedure TForm1.LocationSensor1LocationChanged(Sender: TObject;
  const OldLocation, NewLocation: TLocationCoord2D);
begin
  F1 := NewLocation.Latitude;
  L1 := NewLocation.Longitude;
  WGS_SK(F1, L1, 0);
  SK_Pr(B, L);
  x1 := x;
  y1 := y;

```

```
Label1.Text := FloatToStr(x1);
Label2.Text := FloatToStr(y1);
dX := StrToFloat(Edit2.Text) - x1;
dY := StrToFloat(Edit3.Text) - y1;
Memo1.Lines.Add(FloatToStrF(dX, ffFixed, 11, 3) + ' ' + FloatToStrF(dY,
    ffFixed, 11, 3));
end;
```

Исправляет координаты и отправляет клиенту на ровер.