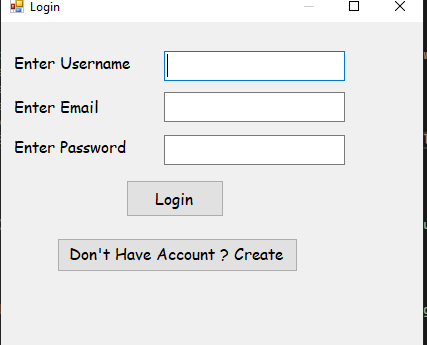
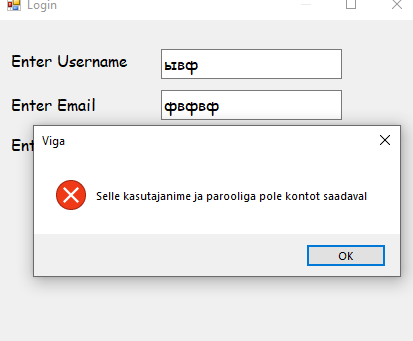
## Registreerimine ja sisselogimine

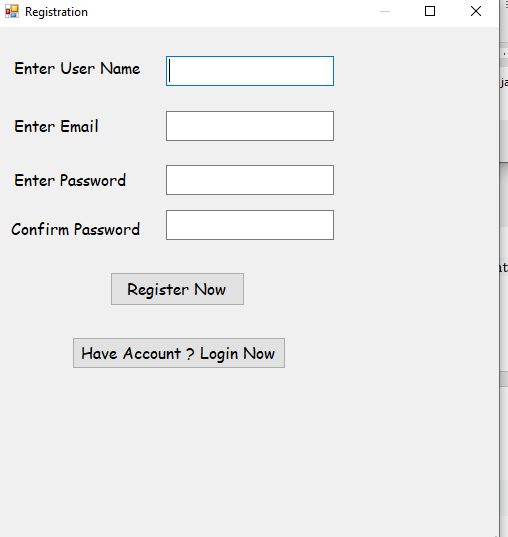
Vormi käivitamisel avaneb sisselogimisvorm, kuhu saab sisse logida olemasoleva kontoga või vajutada registreerimisnuppu ja luua uus kasutaja.



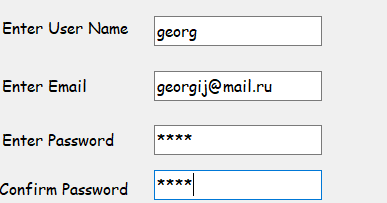
Kui sisestate valeandmed, kuvatakse tõrketeade



klõpsate nuppu Pole kontot, suunatakse teid vormi, kus peate konto looma

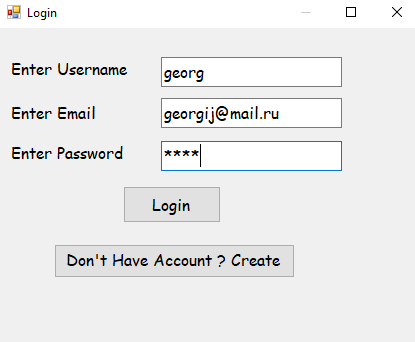


Kui ma kasutaja registreerin, salvestatakse see andmebaasi





Ja parast seda ma sisselogin minu menüsse



Registration.cs

public partial class Registration : Form

{

SqlCommand cmd;

SqlConnection cn;

SqlDataReader dr;

public Registration()

{

InitializeComponent();

}

private void Registration\_Load(object sender, EventArgs e)

{

cn = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\opilane\source\repos\TTT\RegistrationAndLogin\Database.mdf;Integrated Security=True");

cn.Open();

}

private void BtnRegister\_Click(object sender, EventArgs e)

{

if (txtconfirmpassword.Text != string.Empty || txtpassword.Text != string.Empty || txtusername.Text != string.Empty || txtemail.Text != string.Empty)

{

if (txtpassword.Text == txtconfirmpassword.Text)

{

cmd = new SqlCommand("select \* from LoginTable where username='" + txtusername.Text + "'", cn);

dr = cmd.ExecuteReader();

if (dr.Read())

{

dr.Close();

MessageBox.Show("Kasutajanimi on juba olemas, proovige teist ", "Viga", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else

{

dr.Close();

cmd = new SqlCommand("insert into LoginTable values(@username,@password,@email)", cn);

cmd.Parameters.AddWithValue("username", txtusername.Text);

cmd.Parameters.AddWithValue("password", txtpassword.Text);

cmd.Parameters.AddWithValue("email", txtemail.Text);

cmd.ExecuteNonQuery();

MessageBox.Show("Teie konto on loodud. Palun logi kohe sisse.", "Valmis", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

else

{

MessageBox.Show("Sisestage kaks identset parooli ", "Viga", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

else

{

MessageBox.Show("Palun sisestage väärtus kõigile väljadele.", "Viga", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void Button1\_Click(object sender, EventArgs e)

{

this.Hide();

Login login = new Login();

login.ShowDialog();

}

}

Login.cs

public partial class Login : Form

{

SqlCommand cmd;

SqlConnection cn;

SqlDataReader dr;

//

public Login()

{

InitializeComponent();

}

private void Login\_Load(object sender, EventArgs e)

{

cn = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\opilane\source\repos\TTT\RegistrationAndLogin\Database.mdf;Integrated Security=True");

cn.Open();

}

private void Btnregister\_Click(object sender, EventArgs e)

{

this.Hide();

Registration registration = new Registration();

registration.ShowDialog();

}

private void BtnLogin\_Click(object sender, EventArgs e)

{

if (txtpassword.Text != string.Empty || txtusername.Text != string.Empty || txtemail.Text != string.Empty)

{

cmd = new SqlCommand("select \* from LoginTable where username='" + txtusername.Text + "' and email='" + txtemail.Text + "' and password='" +txtpassword.Text+"'", cn);

dr = cmd.ExecuteReader();

if (dr.Read())

{

dr.Close();

this.Hide();

Starting starting = new Starting();

starting.ShowDialog();

}

else

{

dr.Close();

MessageBox.Show("Selle kasutajanime ja parooliga pole kontot saadaval ", "Viga", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

else

{

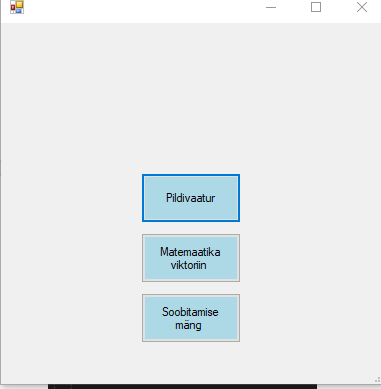
MessageBox.Show("Palun sisestage väärtus kõigile väljadele.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);

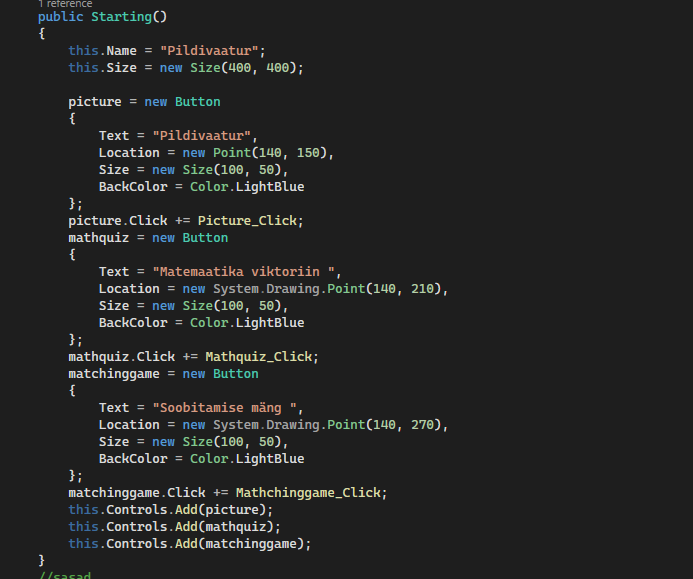
}

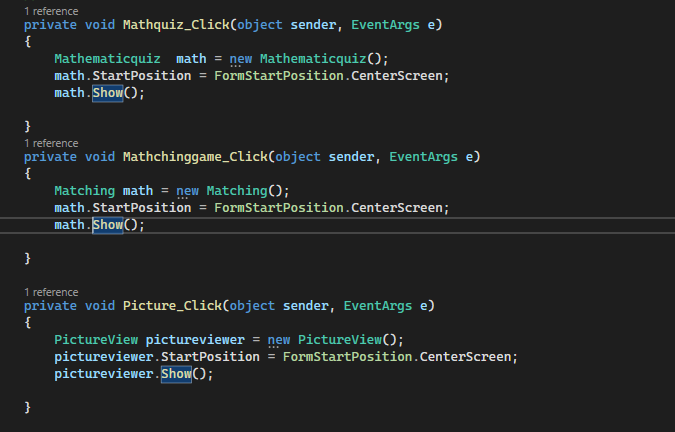
}

}

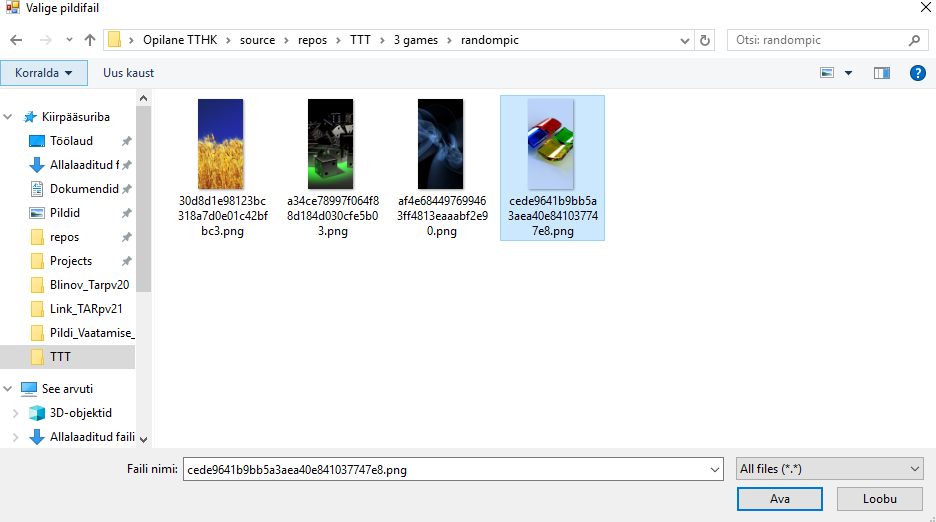
Menüü mangud

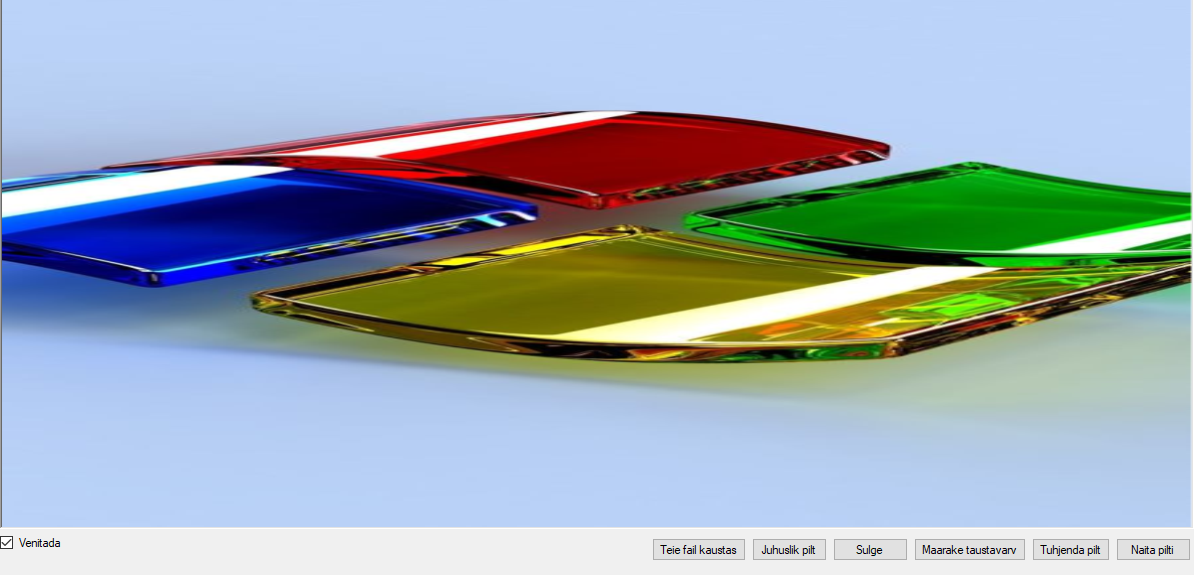




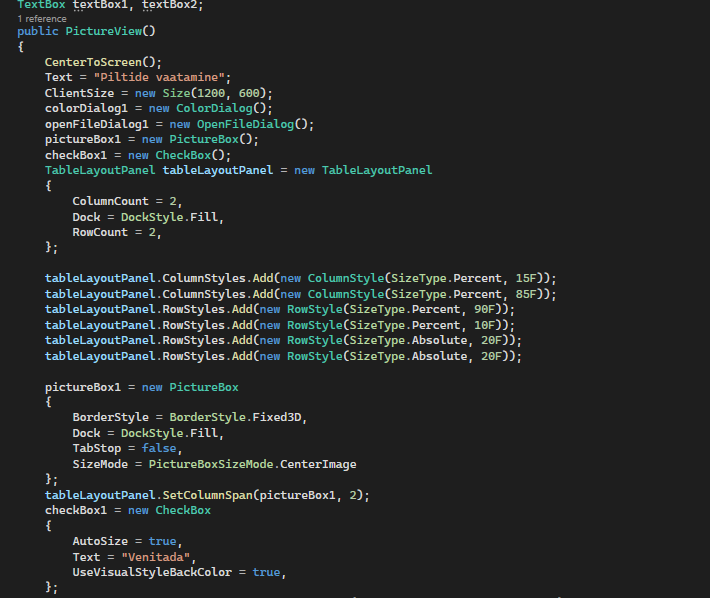


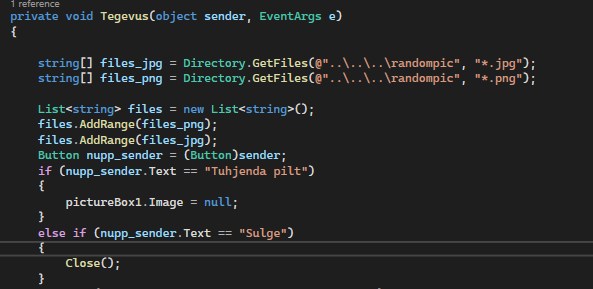


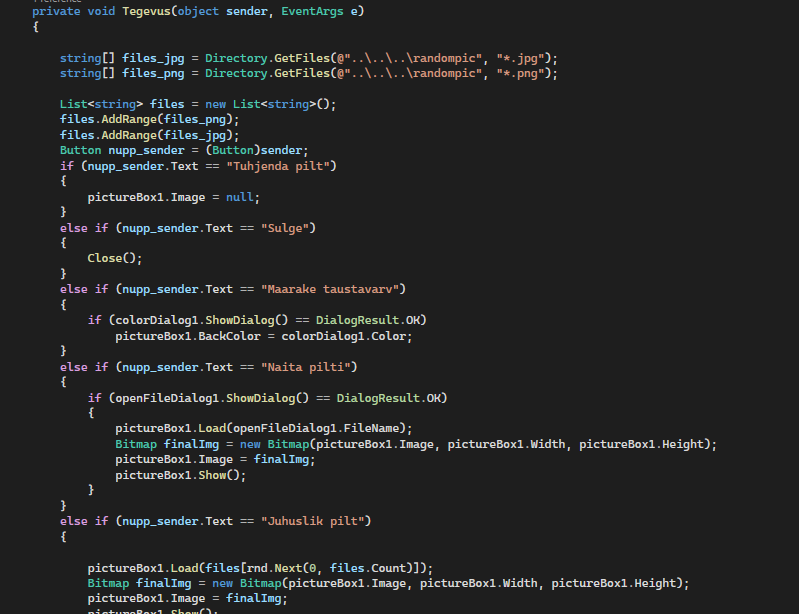




Kui klõpsate kustutamisnupul, siis see kustutatakse; kui klõpsate sulgemisnupul, suletakse kõik







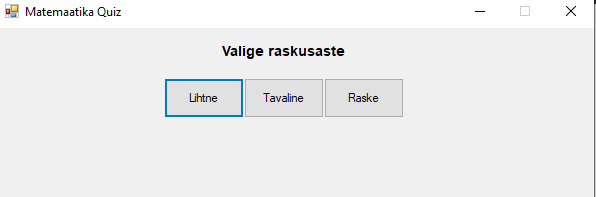
Seal me võime võta juhuslikku pilti kaustast



Matemaatika viktoriin

Kui klõpsan nuppu matematiika viktoriin algab töötada programmi

Seal on kolm raskused



public void Timer\_Tick(object sender, EventArgs e)

{

tik--;

timelabel.Text = "Taimer: " + tik.ToString();

if (tik == 0)

{

timer.Stop();

if (this.difficult == "Lihtne")

{

if (score < 2)

{

MessageBox.Show("Halb", "Hinne");

}

else if (score >= 2 && score < 5)

{

MessageBox.Show("Keskmine", "Hinne");

}

else if (score >= 5)

{

MessageBox.Show("Hea", "Hinne");

}

}

else if (this.difficult == "Tavaline")

{

if (score < 2)

{

MessageBox.Show("Halb", "Hinne");

}

if (score >= 2 && score < 4)

{

MessageBox.Show("Keskmine", "Hinne");

}

if (score >= 4)

{

MessageBox.Show("Hea", "Hinne");

}

}

else if (this.difficult == "Raske")

{

if (score == 0)

{

MessageBox.Show("Halb", "Hinne");

}

else if (score < 2 && score != 0)

{

MessageBox.Show("Pole halb", "Hinne");

}

else if (score >= 2 && score < 4)

{

MessageBox.Show("Üle keskmise", "Hinne");

}

else if (score >= 4)

{

MessageBox.Show("Väga hästi", "Hinne");

}

}

}

}

public void difficultChoice(object sender, EventArgs e)

{

this.Controls.Clear();

Button nupp\_sender = (Button)sender;

if (nupp\_sender.Text == "Lihtne")

{

x = 20;

y = 2;

difficult = "Lihtne";

}

else if (nupp\_sender.Text == "Tavaline")

{

x = 30;

y = 3;

difficult = "Tavaline";

}

else if (nupp\_sender.Text == "Raske")

{

x = 50;

y = 5;

difficult = "Raske";

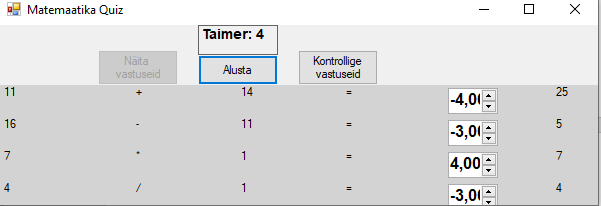
}

matem = new Mathematicquiz(x, y, difficult);

matem.Show();

}

Me valime lihtne ja kui me teeme valesti viktoriini siis me võime kontrollida vastused



private void showTrueAns(object sender, EventArgs e)

{

int[] ans = new int[4];

ans[0] = intnum[0] + intnum2[0];

ans[1] = intnum[1] - intnum2[1];

ans[2] = intnum[2] \* intnum2[2];

ans[3] = intnum[3] / intnum2[3];

for (int i = 0; i < 4; i++)

{

Label l = new Label { Text = ans[i].ToString() };

tableLayoutPanel.Controls.Add(l, 6, i);

}

showAns.Enabled = false;

}

Samuti fikseerime tulemused

private void MathScoreTofile(string mark, int score, string difficult) //Meetod kirjutab faili vastatud testide arvu, raskusastme ja hinde

{

StreamWriter to\_file = new StreamWriter(@"..\..\..\Score2.txt", true);

to\_file.Write($"Vastatud testide arv -> {score.ToString()}" + " -- " + $"Keerukus -> {difficult}" + " -- " + $"Hinne -> {mark}" + "\n");

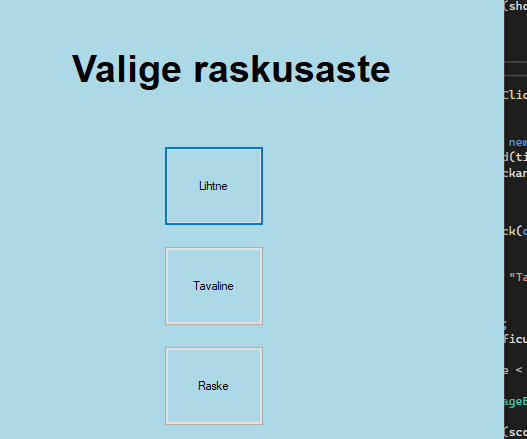
to\_file.Close();

}

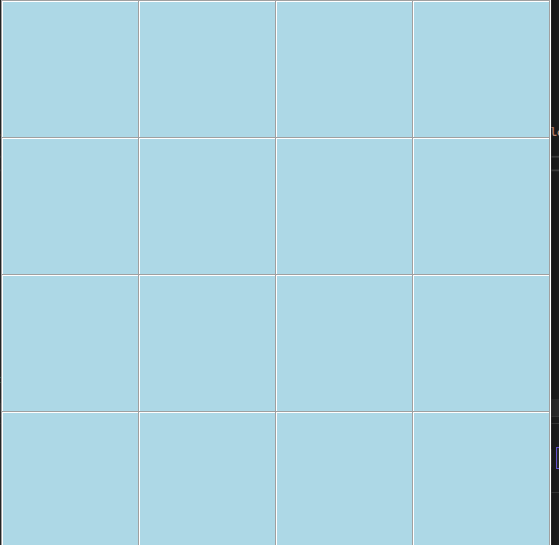
Matching.cs

Kui klõpsan nuppu matematiika viktoriin algab töötada programmi

Seal on kolm raskused



Me valime Tavaline raskus



Seal on salvestatud tulemused failis score

void FailedScoreTofile(int score)

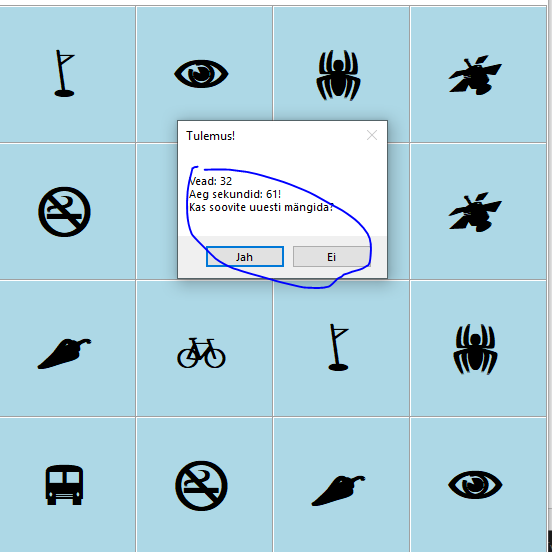
{

StreamWriter to\_file = new StreamWriter(@"..\..\..\Score.txt", true);

to\_file.Write(score.ToString() + " -- " + tik.ToString() + "sek" + "\n");

to\_file.Close();

}



if (nupp\_sender.Text == "Lihtne")

{

new Matching(4, 3, icons, tableLayoutPanel);

}

else if (nupp\_sender.Text == "Tavaline")

{

new Matching(4, 4, icons\_2, tableLayoutPanel);

}

else if (nupp\_sender.Text == "Raske")

{

new Matching(5, 4, icons\_3, tableLayoutPanel);

}

}

List<string> icons = new List<string>()

{

"!", "!", "N", "N", "r", "r",

"b", "b", "v", "v", "~", "~",

};

List<string> icons\_2 = new List<string>()

{

"!", "!", "N", "N", ",", ",", "k", "k",

"b", "b", "v", "v", "w", "w", "z", "z"

};

List<string> icons\_3 = new List<string>()

{

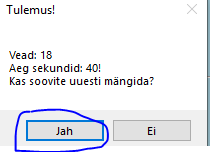
"!", "!", "N", "N", ",", ",", "k", "k", "`", "`",

"b", "b", "v", "v", "w", "w", "z", "z", "f", "f"

};

string[] buttonstext = { "Lihtne", "Tavaline", "Raske" };

Me võime taaskaivita programmi



void restarGame()

{

if (MessageBox.Show($"Vead: {score.ToString()}\nAeg sekundid: {tik.ToString()}!\nKas soovite uuesti mängida?", "Tulemus!", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

Application.Restart();

Environment.Exit(0);

}

else

{

Application.Exit();

}

}