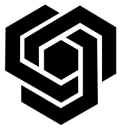
 **ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ** 

Програмни Среди

КУРСОВА РАБОТА:

**Да се направи игра тип „Сделка или не“.**

**Автоматично (и случайно разпределение на определени суми в кутиите).**

**Да се направи елементарен алгоритъм за предложенията на банкера.**

Разработил: Михайло Апостолов Одобрил: ........................... Ф№: 123215006, група 39 / ас. Веселин Божинов /

Описание на реализацията и кода:

Класа BoxData съдържа два списъка от обекти тип Box : CurentBoxes и CurentBoxesR, и два метода FillBoxesInOrder() и FillBoxesRandomly().

Метода FillBoxesInOrder() попълва списъка от обекти “CurentBoxes” с стойностите които трябва да са в кутиите, но го прави така че да стойностите са подредени от най-малка до най-голяма.

static private void FillBoxesInOrder()

{

string[] BoxValues = new string[22] { "0.10","0.20","0.50","1","Предмет","10","20","50","100","200",

"300", "750","1,000","2,500","5,000","7,500","10,000","12,500","15,000","25,000","50,000","100,000"};

int loop = 1;

for (int i = 0; i < BoxValues.Length; i++)

{

\_CurentBoxes.Add(new Box()

{

boxNumber = loop,

boxValue = BoxValues[i]

}

);

loop++; }

Метода FillBoxesInRandomly () попълва списъка от обекти “CurentBoxesR” с стойностите които трябва да са в кутиите, но го прави така че стойностите в кутиите да са распределени по случаен начин.

static private void FillBoxesRandomly()

{

FillBoxesInOrder();

int newBoxIndex = 1;

var rnd = new Random(DateTime.Now.Millisecond);

while (\_CurentBoxes.Count != 0)

{

int index = rnd.Next(0, \_CurentBoxes.Count);

\_CurentBoxesR.Add(new Box()

{

boxNumber = newBoxIndex,

boxValue = \_CurentBoxes[index].boxValue

} );

\_CurentBoxes.Remove(\_CurentBoxes[index]);

newBoxIndex++; }

MainWindow.xaml – е прозореца чрез който се стартира играта.

Бутона в МainWindow „Вход в играта“ стартира нова инстанция на прозореца Game.

private void LogInBtn\_Click(object sender, RoutedEventArgs e)

{

Game ng = new Game();

ng.Show();

ng.Start();

}

Game.xaml – е прозореца на самата игра. Той съдържа 22 бутона които представляват кутиите, 2 бутона за „Сделка!“ и “Няма Сделка!“, бутон за смяна на кутиите, listBox за следене на стъпките.

Метода SetBoxes()- поставя полето „Content“ на бутоните представляващи кутиите, така че да бъдат еднакви на полето „boxNumber“ от класа Box.

private void SetBoxes()

{

Box22.IsEnabled = false;

foreach (Button btn in GameGrid.Children)

{

if (btn.Name.StartsWith("Box"))

{

btn.Content = bo[GameGrid.Children.IndexOf(btn)].boxNumber;

ButonBoxes.Add(btn);

}

}

}

Метода Start()- подава към лист бокса “steps” първата стъпка която потребителя трябва да направи, и прави невидими/неактивни всички в момента не нужни бутони и контроли.

public void Start()

{

ListBoxItem rule1 = new ListBoxItem

{

Content = "Отворете 5 кутии по избор."

};

steps.Items.Add(rule1);

Smqna.Visibility = Visibility.Hidden;

swapIndex.Visibility = Visibility.Hidden;

offerLable.Visibility = Visibility.Hidden;

Zdelka.IsEnabled = false;

NqmaZdelka.IsEnabled = false; }

Метода SwapBoxes(int bts)- приема номера на кутията кято потребителя иска да смени с своята, и взима бутоните представляващи кутиите и проверява дали съществува бутон с номер еднакъв на подадения от потребителя и ако съществува, променя кутията на потребителя с избраната от него нова кутия.

private void SwapBoxes(int boxToSwap)

{

foreach(Button sb in ButonBoxes)

{

if (sb.IsEnabled)

{

if (int.Parse(sb.Content.ToString()) == boxToSwap)

{

Smqna.Visibility = Visibility.Hidden;

swapIndex.Visibility = Visibility.Hidden;

NqmaZdelka.IsEnabled = false;

ListBoxItem SwapLog = new ListBoxItem

{

content = ">>Сменихте кутия: " + Box22.Content + " за кутия: " + sb.Content

};

steps.Items.Add(SwapLog);

steps.ScrollIntoView(SwapLog);

int boxIndex = ButonBoxes.IndexOf(sb);

Box temp = bo[21];

bo[21] = bo[boxIndex];

bo[boxIndex] = temp;

Button tempbtn = new Button()

{

Content = Box22.Content

};

Box22.Content = sb.Content;

sb.Content = tempbtn.Content;

}

}

}

}

Метода GamePlay()- се извиква всеки път когато се отвори нова кутия.Той проверява боря на отворените кутии. За 5,9,12,14,16,18,20 и 21 отворени кутии дава до знание на потребителя колко кутии е отворил до сега и за всеки case извиква метода BankOffer() .

private void GamePlay()

{

Zdelka.IsEnabled = false;

NqmaZdelka.IsEnabled = false;

Smqna.Visibility = Visibility.Hidden;

swapIndex.Visibility = Visibility.Hidden;

offerLable.Visibility = Visibility.Hidden;

offerLable.Content = "";

switch (BoxesOpened())

{

case 5:

ListBoxItem fiveOpened = new ListBoxItem

{

Content = "Отворихте 5 кутии. Предстои оферта на Банкера"

};

steps.Items.Add(fiveOpened);

BankOffer();

break;

case 9:

ListBoxItem nineOpened = new ListBoxItem

{

Content = "Отворихте 9 кутии. Предстои оферта на Банкера"

};

steps.Items.Add(nineOpened);

BankOffer();

break;

case 12:

ListBoxItem twelveOpened = new ListBoxItem

{

Content = "Отворихте 12 кутии. Предстои оферта на Банкера"

};

steps.Items.Add(twelveOpened);

BankOffer();

break;

case 14:

ListBoxItem fourteenOpened = new ListBoxItem

{

Content = "Отворихте 14 кутии. Предстои оферта на Банкера"

};

steps.Items.Add(fourteenOpened);

BankOffer();

break;

case 16:

ListBoxItem sxsOpened = new ListBoxItem

{

Content = "Отворихте 16 кутии. Предстои оферта на Банкера"

};

steps.Items.Add(sxsOpened);

BankOffer();

break;

case 18:

ListBoxItem eighteenOpened = new ListBoxItem

{

Content = "Отворихте 18 кутии. Предстои оферта на Банкера"

};

steps.Items.Add(eighteenOpened);

BankOffer();

break;

case 20:

ListBoxItem twnOpened = new ListBoxItem

{

Content = "Отворихте 20 кутии. Предстои оферта на Банкера"

};

steps.Items.Add(twnOpened);

BankOffer();

break;

case 21:

if(bo[21].boxValue == "Предмет")

{

MessageBox.Show("Честито вие спечелихте: " + bo[21].boxValue + "!");

Box22.Content = bo[21].boxValue;

}

else

{

MessageBox.Show("Честито вие спечелихте: "+bo[21].boxValue + "лева!");

Box22.Content = bo[21].boxValue;

}

ListBoxItem endGame = new ListBoxItem

{

Content = "Честитоо! \n Вие завършихте играта. "

};

steps.Items.Add(endGame);

steps.ScrollIntoView(endGame);

MessageBox.Show("Започнете нова Игра.");

this.Close();

break;

default:

break;

}

}

Метода BankOffer()- представлява елементарен алгоритъм за предложения на банкера. По случен принцип се избира дали потребителя ще добие парична оферта или щанс да си смени кутията.

private void BankOffer()

{

NqmaZdelka.IsEnabled = true;

Random rnd = new Random(DateTime.Now.Millisecond);

int ofer = rnd.Next(1, 6);

int[] pari = new int[5] { 200, 500, 800, 1000, 3000 };

switch (ofer)

{

case 1:

Smqna.Visibility = Visibility.Visible;

swapIndex.Visibility = Visibility.Visible;

ListBoxItem Ofer1 = new ListBoxItem

{

Content = ">>Банкера ви предлага смяна на кутиите."

};

steps.Items.Add(Ofer1);

steps.ScrollIntoView(Ofer1);

break;

case 2:

Zdelka.IsEnabled = true;

ListBoxItem Ofer2 = new ListBoxItem

{

Content = ">>Банкера ви предлага парична сума."

};

steps.Items.Add(Ofer2);

steps.ScrollIntoView(Ofer2);

offerLable.Visibility = Visibility.Visible;

int p = rnd.Next(5);

offerLable.Content = "Банкера ви предлага" + pari[p]+" Лева!";

BOInt = pari[p];

break;

case 3:

Zdelka.IsEnabled = true;

ListBoxItem Ofer3 = new ListBoxItem

{

Content = ">>Банкера ви предлага парична сума."

};

steps.Items.Add(Ofer3);

steps.ScrollIntoView(Ofer3);

offerLable.Visibility = Visibility.Visible;

int q = rnd.Next(5);

offerLable.Content = "Банкера ви предлага" + pari[q] +" Лева!";

BOInt = pari[q];

break;

case 4:

Smqna.Visibility = Visibility.Visible;

swapIndex.Visibility = Visibility.Visible;

ListBoxItem Ofer4 = new ListBoxItem

{

Content = ">>Банкера ви предлага смяна на кутиите."

};

steps.Items.Add(Ofer4);

steps.ScrollIntoView(Ofer4);

break;

case 5:

Smqna.Visibility = Visibility.Visible;

swapIndex.Visibility = Visibility.Visible;

ListBoxItem Ofer5 = new ListBoxItem

{

Content = ">>Банкера ви предлага смяна на кутиите."

};

steps.Items.Add(Ofer5);

steps.ScrollIntoView(Ofer5);

break;

default:

break;

}

}

Метода BoxesOpened()- върща като резултат броя на отворените кутии.

private int BoxesOpened()

{

int openBoxes = -1;

foreach (Button bt in ButonBoxes)

{

if(bt.IsEnabled == false)

{

openBoxes++;

}

}

return openBoxes;

}