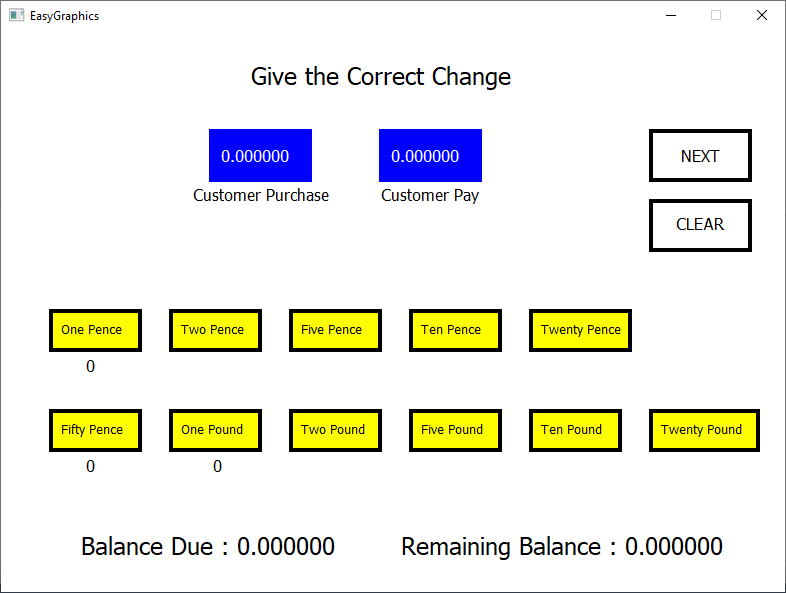
This week, we are going to continue on developing the game Give the Correct Change.

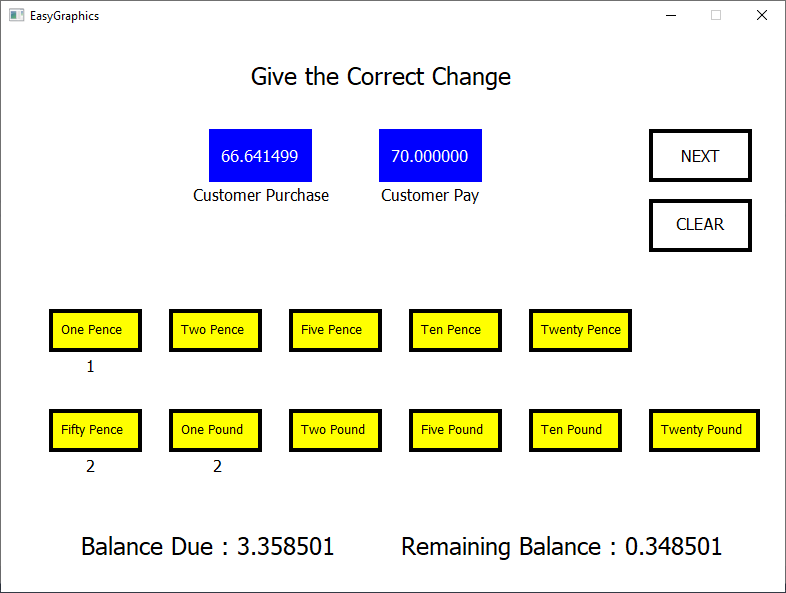
We need to write some functions for the followings,

* to compute the balance due
* to get the remaining balance



Balance due = Customer Pay – Customer Purchase

Remaining Balance = Balance Due – Sum of the Denominations



0.34 = 3.35 – (0.01 x 1) – (0.50 x 2) – (1.0 x 2)

**Compute the Balance Due**

Firstly, declare all the global variables and prototypes that needed. You might need to have three array variables such as,

int denomCount [11] = {0,0,0,0,0,0,0,0,0,0,0};

double denom[11] = { 0.01,0.02,0.05,0.1,0.2,0.5,1.0,2.0,5.0,10.0,20.0};

double amounts[2] = { 0.00,0.00 }; // index 0 is for purchase amount, index 1 is for pay amount

You also need to declare these variables,

double balance = 0.0, balanceAnswer = 0.0;

Also, these prototypes for the functions,

double getBalance();

For the remaining balance funtion

double checkRemainingBalance();

Write all these codes in your ***DrawingTool.h*** file.

Assuming that you have designed the GUI for the denominations and texts for the balance due and remaining balance, write these codes (in ***DrawingTool.cpp***) to the both values.

double DrawingTool::getBalance()

{

return balance = amounts[1] - amounts[0];

}

Why the amount from index 1 minus index 0?

Now, display the value on the GUI,

Create your own function to draw a text

void DrawingTool::balanceDue()

{

setTextColour(BLACK);

setFont(18, L"Tahoma");

Call the getBalance function and assign to a double local variable

string text = "Balance Due : ";

double due = getBalance();

text += to\_string(due);

drawText(text.c\_str(), 80, 500);

}

Study these codes. What it does?

Don’t forget to call **balanceDue()** function from **onDraw()**. Otherwise, you are now drawing the text on the canvas.

**Compute the Remaining Balance**

Earlier on, you have declared all the variables. Now, write the function to get the value for the remaining balance. The remaining balance value will increase (or decrease) when the user selects (deselect) the denominations. The user gets the answer right when the remaining balance is 0.00!

In your ***DrawingTool.cpp*** file, write this function,

double DrawingTool::checkRemainingBalance()

This is a local variable being used in this function only

{

double answer=0;

for (int i=0; i<11; i++)

{

answer += (denomCount[i] \* denom[i]);

}

Multiply the count of the denominations and the value of the denominations

return balanceAnswer = balance- answer;

}

Variable assignment to the global variable

Now display the value on the GUI,

Create your own function to draw a text

void DrawingTool::answerText()

{

setTextColour(BLACK);

setFont(18, L"Tahoma");

The function checkRemainingBalance is called here

string text = "Remaining Balance : ";

double bal = checkRemainingBalance();

text += to\_string(bal);

drawText(text.c\_str(), 400, 500);

}

**Discussion**

When the user clicks on the Next rectangle, the game will continue with next values for Customer Purchase and Customer Pay. Since the count of the denominations are stored in an array, you need to set the values to 0 for the next challenge.

