cs102 lab 2

Specification

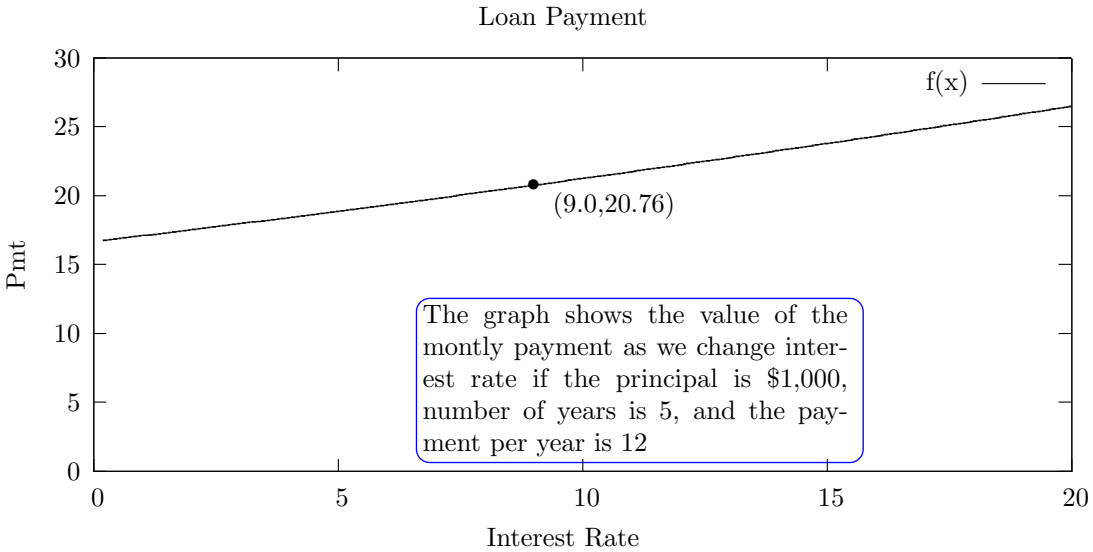
The "Payment" refers to the amount of each montly payment

$$payment = \frac{intRate \times \frac{principal}{payPerYear}}{1 - (\frac{intRate}{payPerYear} + 1)^{-payPerYear \times numYears}}$$

For example, the payment for interest rate of .09, principle of 1000, with 12 payments per year, and 5 years for loan.

$$payment = \frac{.09 \times \frac{1000}{12}}{1 - (\frac{.09}{12} + 1)^{-12 \times 5}}$$

Analysis


Loan Payment
30
25
20
15
10
5
0
Pmt
0
5
10
15
20
Interest Rate
f(x)
(9.0,20.76)
The graph shows the value of the montly payment as we change interest rate if the principal is $1,000, number of years is 5, and the payment per year is 12

Design

- inputs: Principle, Interest rate, number of Payments per Year, number of Years
- inputs are put in through a Graphic User Interface, with 4 text boxes, one for each variable
- outputs: monthly payment
- outputs are displayed in an unalterable text box
- process (convert inputs to outputs)
  - ask user to enter the all the information, in the respective text box in the GUI GUI is short for Graphic User Interface
  - Press Calculate
  - then call function f (which calls function pmt which use the formula to calculate the monthly payment)
  - use "RND" function to round the value to cents, and then display in the appropriate text box

Implementation lab.h

List of all Variables and functions

```
#include <config.h>
#include <cmath>
#include <FL/Fl_Cairo_Window.H>
#include <FL/Fl_Value_Input.H>
#include <FL/Fl_Value_Output.H>
#include <FL/Fl_Button.H>
#include <FL/Fl_Box.H>
#include <FL/Fl_PNG_Image.H>

double f(double r, double a, double ppy, double n);
double pmt(double r,double a,double ppy,double n);
Fl_Cairo_Window * make_window();
void cb_Calculate(Fl_Button*,void*);
extern Fl_Cairo_Window * cw;
extern const int width;
extern const int height;
extern Fl_Button * b;
extern Fl_Box * g;
extern Fl_Value_Output * p;
extern Fl_Value_Input *r;
extern Fl_Value_Input *a;
extern Fl_Value_Input *ppy;
extern Fl_Value_Input *n;
```

Implementation labgui.cpp

Declarations of all FLTK variables

```
#include "lab.h"
Fl_Cairo_Window * cw;
Fl_Value_Input * r;
Fl_Value_Input * a;
Fl_Value_Input * ppy;
Fl_Value_Input * n;
Fl_Value_Output * p;
Fl_Button * b;
Fl_Box * g;
const int width = 300; //number of pixels of width of the window
const int height = 300; // same as width but for height
```

Implementation lab.cpp

```
#include <iostream>
#include <iomanip>
#include "lab.h"
using namespace std;
int main ()
{

    make_window()->show();
    Fl::run();
    return 0;
}
double f(double r, double a, double ppy, double n)
{
    return pmt(r,a,ppy,n);
}
double pmt(double r, double a, double ppy, double n)
{
    return ((r/100.0) * (a/ppy)) / (1-pow((r/100.0/ppy+1),-(ppy*n)));
}
```

Make Window is define on clabgui2

The function "PMT" uses 4 double variables. a=principle,r=interest rate, ppy=payments per year, n=number of years. Then it inputs each variable into the monthly payment equation returning the payment value.

Implementation clabgui1.cpp

```
#include "lab.h"
double rnd(double d)
{
    d=d*100;
    d=std::round(d);
    d=d/100;
    return d;
}

void cb_Calculate(Fl_Button*,void*)
{
    p->value(rnd(f(r->value(),a->value(),ppy->value(),n->value())));
}
```

Rounding works by:

- 1) Multiplying by 100
- 2) Rounding the number to the nearest whole integer
- 3) Dividing by 100 to return to dollars and cents

p's value is composition of the "rnd" function of the "f" function of the values of r, p, ppy, and n.

Implementation clabgui2.cpp

The Make Window Function on its own in it's entirety

The name of a cairo text box is repective to the variable that the input is saved to.

```
#include "lab.h"
Fl_Cairo_Window * make_window(){
    cw = new Fl_Cairo_Window(width,300);
    cw->label("Lab 2:Loan Payment Calculator");
    cw->color(fl_rgb_color(121,152,182));
    a = new Fl_Value_Input(.6*width,.05*height,.25*width, .1*height);
    //number are how far in, how far down, how wide type, how tall type
    a->label("Principal:");
    r = new Fl_Value_Input(.6*width,.15*height,.25*width, .1*height);
    r->label("Interest Rate (9% = 9):");
    ppy = new Fl_Value_Input(.6*width,.25*height,.25*width, .1*height);
    ppy->label("# of Payments per Year:");
    n = new Fl_Value_Input(.6*width,.35*height,.25*width, .1*height);
    n->label("# of Year:");
    p = new Fl_Value_Output(.6*width,.75*height,.25*width, .1*height);
    p->label("Monthly Payment:");
    b = new Fl_Button(.6*width,.55*height,.25*width, .1*height);
    b->label("Calculate");
    b->color(FL_BLUE); b->labelcolor(FL_WHITE);
    b->callback((Fl_Callback*)cb_Calculate);
    g = new Fl_Box(FL_FLAT_BOX,.25*width,.535*height,64,64,"");
    g->color(fl_rgb_color(121,152,182));
    g->image(new Fl_PNG_Image("loan.png")); return cw;}
```

Test

- User can enter any data they desire
- If user enters principal of $1,000, interest rate of 9%, payments per year is 12, and number of years is 5:
- If user enters principal of $10000, interest rate of 3%, payments per year is 6, and number of years is 4:

The figures show the functions ability to round up and down