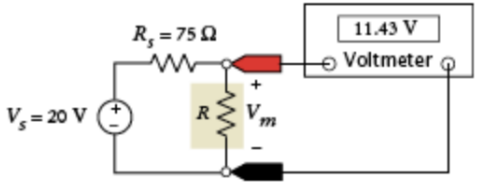
Specification

INSTRUCTIONS

The electric circuit shown below is designed to measure the temperature of the gas in a chamber.



The resistor R represents a temperature sensor enclosed in the chamber. The resistance R, in Ω, is related to the temperature T, in °C, by the equation

$$R = R_0 + kT$$

In this device, assume $R_0 = 100\ \Omega$ and $k = 0.5$. The voltmeter displays the value of the voltage, V_m, across the s This voltage V_m indicates the temperature, T, of the gas according to the equation

$$T = \frac{R}{k} - \frac{R_0}{k} = \frac{R_s}{k}\frac{V_m}{V_s - V_m} - \frac{R_0}{k}$$

Suppose the voltmeter voltage is constrained to the range $V_{\min} = 12$ volts $\leq V_m \leq V_{\max} = 18$ volts. Write a progr that accepts a value of V_m and checks that it's between 12 and 18. The program should return the gas temperat degrees Celsius when V_m is between 12 and 18 and an error message when it isn't.

The programe will show a window to the users and tell them to enter the voltage and then output the

gas temperature as the question tells us.

Analysis

Inputs The voltage $V_m, R_0 = 100\Omega, k = 0.5, R_s = 75\ \Omega, V_s = 20V$

Process Calculate temperature T from input V_m where

$$T = \frac{R_s}{k} \times \frac{V_m}{V_{s-V_m}} - \frac{R_0}{k}$$

where $R_0 = 100\Omega, k = 0.5, R_s = 75\ \Omega, V_s = 20V$

Outputs The temperature in Celsius or error message

Design

- Create constants for the question. $R_0 = 100\Omega, k = 0.5, R_s = 75\ \Omega, V_s = 20V$
- As the question tells us,we should use the equation, which is shown .So we should use the following equation in our programe to calculate the gas temperature. $T = \frac{Rs*Vm}{k*(Vs-Vm)} - \frac{R0}{k}$
- What's more,we should test if the user typed a wrong thing in our software.
- So in this case,our code will be execute successfully.

Implementation

```
// cs102 lab
#include <iostream>
#include "labgui.h"
int main()
{
        make_window()->show();
        return Fl::run();
}

// generated by Fast Light User Interface Designer (fluid) version 1.0304

#include "labgui.h"

static void cb_Go(Fl_Button*, void*) {
  double vm = voltage->value(); /* v_m is the variable that carries the value of voltage andit
should between 12 to 18.*/


const double rs=75;
const double vs=20;
const double r0=100;
const double k=0.5;
if(vm<12||vm>18) //std::cout <<"NG"<<std::endl;
{
fl_alert("That voltage is out of range.");
msg->show();
}
else if(vm>=12&&vm<=18)
```

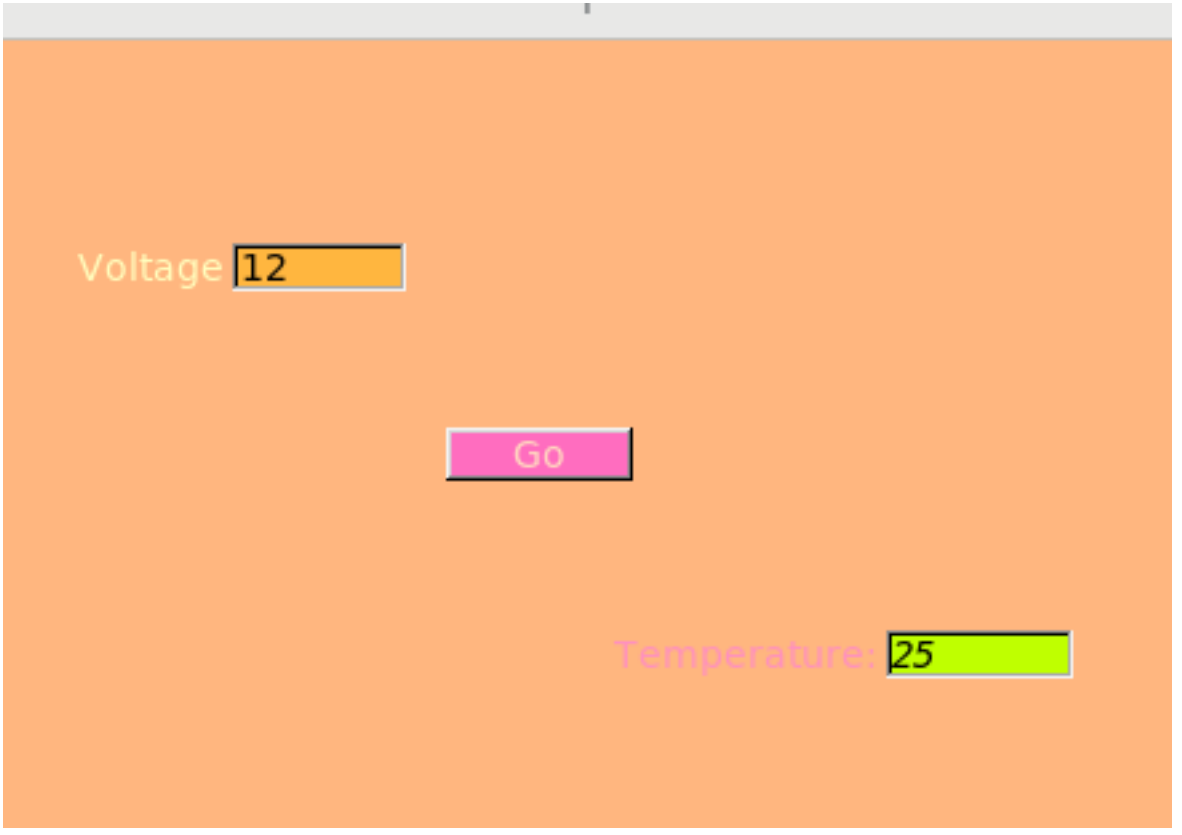
```
{

double t = (rs*vm)/(k*(vs-vm))-r0/k;//t is a variable that carries the value of temperature

temperature->value(t);
};
}

Fl_Value_Input *voltage=(Fl_Value_Input *)0;

Fl_Value_Output *temperature=(Fl_Value_Output *)0;

Fl_Output *msg=(Fl_Output *)0;

Fl_Double_Window* make_window() {
  Fl_Double_Window* w;
  { Fl_Double_Window* o = new Fl_Double_Window(460, 320, "Gas Temperature");
    w = o; if (w) {/* empty */}
    o->color((Fl_Color)173);
    { // calculate correct temp based on voltage
      Fl_Button* o = new Fl_Button(180, 145, 70, 20, "Go");
      o->color((Fl_Color)211);
      o->labelcolor((Fl_Color)214);
      o->callback((Fl_Callback*)cb_Go);
    } // Fl_Button* o
    { voltage = new Fl_Value_Input(100, 76, 65, 18, "Voltage");
      voltage->color((Fl_Color)133);
      voltage->labelcolor((Fl_Color)215);
    } // Fl_Value_Input* voltage
    { temperature = new Fl_Value_Output(345, 221, 70, 18, "Temperature:");
```

```
      temperature->color((Fl_Color)87);
      temperature->labelcolor((Fl_Color)212);
      temperature->textfont(6);
    } // Fl_Value_Output* temperature
    { msg = new Fl_Output(350, 41, 0, 23, "Voltage is out of range");
      msg->labelfont(3);
      msg->labelsize(24);
      msg->labelcolor((Fl_Color)81);
      msg->hide();
    } // Fl_Output* msg
    o->end();
  } // Fl_Double_Window* o
  return w;
}

// generated by Fast Light User Interface Designer (fluid) version 1.0304

#ifndef labgui_h
#define labgui_h
#include <FL/Fl.H>
#include <FL/Fl_Double_Window.H>
#include <FL/Fl_Button.H>
#include <iostream>
#include <FL/fl_ask.H>
using namespace std;
#include <FL/Fl_Value_Input.H>
extern Fl_Value_Input *voltage;
#include <FL/Fl_Value_Output.H>
extern Fl_Value_Output *temperature;
#include <FL/Fl_Output.H>
```

```
extern Fl_Output *msg;
Fl_Double_Window* make_window();
#endif
```

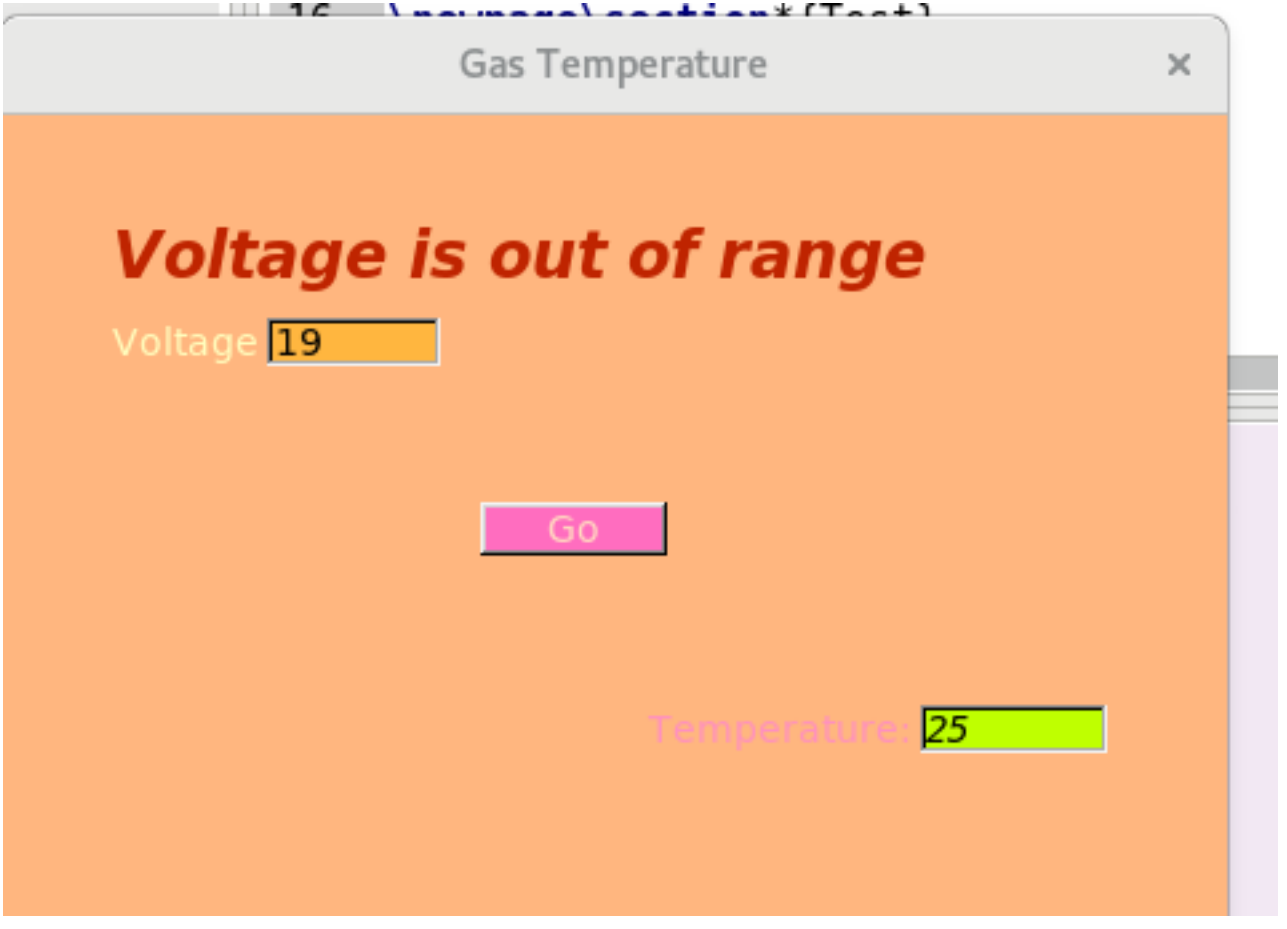
Test

Testcase 1



- So when our input is 12,our output must be 25,which is exactly correct.
- like $25 = \frac{75*12}{0.5*(20-12)} - \frac{100}{0.5}$
- Since our output is exactly 25,our testcase1 is passed.

Testcase 2


Gas Temperature
Voltage is out of range
Voltage 19
Go
Temperature: 25

- So when our input is 19,our output must be the message that tells user the voltage is out of range.
- Since our output is exactly what we need as the picture shows,our testcase2 is passed.