

Lab:Ch9c

1	Description	1
2	The Problem	1
3	Error	2
4	Source Code	2
	4.1 Lab9c.cxx	3
	4.2 Lab9c.h	5
5	Tex File	6
	Thu Pham	
	Profs. Topham	
	CS116	

1 Description

- For this lab, we need to create a class Microwave into the GUI.
- We also implement four functions which are get time, increase time, switch power, and reset.
- We want the user to control the microwave and let them decide what they want.
- We need three buttons total in the GUI one for start, one for add 30 sec, and the last one is for power switch.
- We want the output to display the minutes that remaining and begin with.

2 The Problem

A microwave control panel has four buttons: one for increasing the time by 30 seconds, one for switching between power levels 1 and 2, a reset button, and a start button. Implement a class that simulates the microwave, with a member function for each button. The member function for the start button should print a message "Cooking for ... seconds at level ...".

If this were a text program--only writing to terminal--the output might look like this:

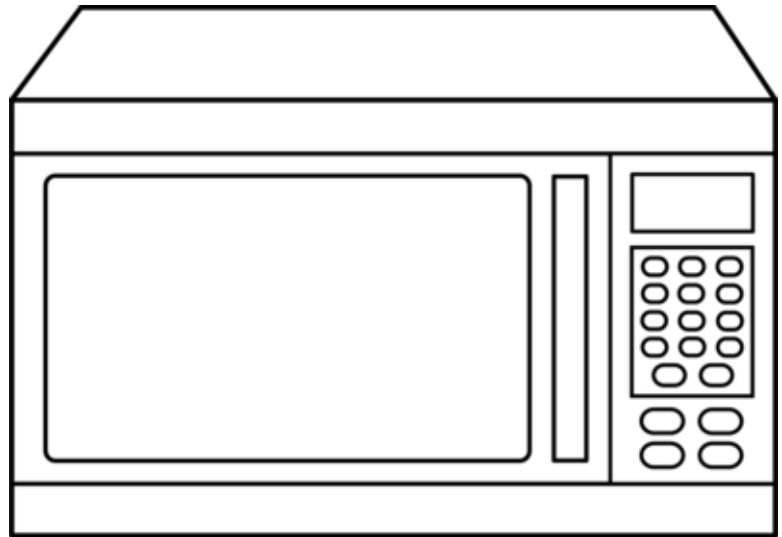
```
Increasing time to 120 seconds
Increasing power level
Cooking for 120 seconds at level 2
Resetting time to seconds
Cooking for 60 seconds at level 1
```

But this is a GUI program, so all I/O will be in a Window.

- Create the class source code manually or using FLUID
 - `/** Increases the time on the timer by 30 seconds. */`
`void increase_time();`
 - `/** Switches the power level from low to high, or vice versa. */`
`void switch_power();`
`/** Resets the microwave to its initial state. */`
`void reset();`
 - `/** Starts the machine, displaying information about its cooking state. */`
`void start();`
- Create a GUI for improved user interaction using FLTK/Fluid
 - Put image of a Microwave as part of the GUI to make program look better.
 - Use Dynamic memory allocation to create the image in the box
 - Show seconds counting down to finish cooking
 - Display some variation to picture when timer counts down to zero
- **Make movie of testing all the functions of your program.**

submit **pdf**, **mp4**, and **tgz** files will all source code visible in pdf. Also show an image of your GUI as first thing in pdf.

3 The Result



This is the picture of the microwave



This is the picture of the cake

4 Source Code

4.1 Lab9c.cxx

```
1 // generated by Fast Light User Interface Designer (fluid) version 1.0305
2
3 #include "lab9c.h"
4
5 Microwave::Microwave() {
6     printf("in_Constructor\n");
7     time = 0;
8     powerLevel = 1;
9 }
10
11 int Microwave::get_time() const {
12     return time;
13 }
14
15 /**
16  add 30 seconds to the current time
17 */
18 void Microwave::increase_time() {
19     time += 30;
20 }
21
22 void Microwave::switch_power() {
23     if(powerLevel == 1)
24     {
25         powerLevel= 2;
26     }
27     else
28     {
29         powerLevel = 1;
30     }
31 }
32
33 void Microwave::reset() {
34     time = 0;
35     powerLevel = 1;
36 }
37
38 void Microwave::tick() {
39     printf("Tick_Tock\n");
40     time --;
41 }
42
43 int Microwave::get_power() const {
44     return powerLevel;
45 }
46
47 /**
48  The countdown as food cooks
49 */
50 void cook_cb(void* op) {
51     m.tick();
52     int rt = m.get_time();
```

```

49     int power = m.get_power();
50     printf("Remaning_time_is_%d\n", rt);
51     printf("Power_Level_is_%d\n", power);
52     ((Fl_Output*)op)->value(std::to_string(rt).c_str());
53     ((Fl_Output*)pl)->value(std::to_string(power).c_str());
54     if (rt <= 0)
55     {
56         Fl::remove_timeout(cook_cb, op);
57         printf("Timer_turned_off\n");
58     }
59     else
60     {
61         printf("cooking...\n");
62         Fl::repeat_timeout(1, cook_cb, op);
63     }
64 }
65
66 Fl_Double_Window *w=(Fl_Double_Window *)0;
67
68 Fl_Box *b=(Fl_Box *)0;
69
70 Fl_Box *c=(Fl_Box *)0;
71
72 Fl_Output *op=(Fl_Output *)0;
73
74 Fl_Button *s=(Fl_Button *)0;
75
76 static void cb_s(Fl_Button*, void*) {
77     std::cout << "start_pushed" <<std::endl;
78     printf ("Cooking_for_%d_seconds_at_level_%d\n", m.get_time(), m.get_power());
79     Fl::add_timeout(1, cook_cb, op);
80 }
81
82 Fl_Button *a=(Fl_Button *)0;
83
84 static void cb_a(Fl_Button*, void*) {
85     m.increase_time();
86     op->value(std::to_string(m.get_time()).c_str());
87 }
88
89 Fl_Button *p=(Fl_Button *)0;
90
91 static void cb_p(Fl_Button*, void*) {
92     m.switch_power();
93     std::cout << "Power_Switch" << std::endl;
94 }
95
96 Fl_Output *pl=(Fl_Output *)0;
97
98 Fl_Button *r=(Fl_Button *)0;
99
100 static void cb_r(Fl_Button*, void*) {
101     m.reset();

```

```

102 op->value(std::to_string(m.get_time()).c_str());
103 }
104
105 int main(int argc, char **argv) {
106     { w = new Fl_Double_Window(400, 275);
107         { Fl_Box* o = b = new Fl_Box(30, 35, 335, 215);
108             Fl_PNG_Image* b = new Fl_PNG_Image("micro.png");
109             o->image(b);
110         } // Fl_Box* b
111         { Fl_Box* o = c = new Fl_Box(40, 45, 335, 215);
112             Fl_PNG_Image* cake = new Fl_PNG_Image("cake.png");
113             o->image(cake);
114         } // Fl_Box* c
115         { Fl_Box* o = new Fl_Box(310, 82, 90, 173);
116             o->box(FL_DOWN_BOX);
117             o->color(FL_BACKGROUND2_COLOR);
118         } // Fl_Box* o
119         { Fl_Output* o = op = new Fl_Output(320, 90, 70, 25);
120             o->value(std::to_string(m.get_time()).c_str());
121         } // Fl_Output* op
122         { s = new Fl_Button(320, 150, 70, 20, "Start");
123             s->box(FL_ROUNDED_BOX);
124             s->color((Fl_Color)178);
125             s->labelcolor(FL_BACKGROUND2_COLOR);
126             s->callback((Fl_Callback*)cb_s);
127         } // Fl_Button* s
128         { a = new Fl_Button(355, 205, 45, 45, "Add_30_sec");
129             a->box(FL_ROUND_UP_BOX);
130             a->color(FL_DARK_GREEN);
131             a->labelcolor(FL_BACKGROUND2_COLOR);
132             a->callback((Fl_Callback*)cb_a);
133             a->align(Fl_Align(FL_ALIGN_WRAP));
134         } // Fl_Button* a
135         { p = new Fl_Button(310, 205, 45, 45, "Power_Level");
136             p->box(FL_ROUND_UP_BOX);
137             p->color((Fl_Color)73);
138             p->labelcolor(FL_BACKGROUND2_COLOR);
139             p->callback((Fl_Callback*)cb_p);
140             p->align(Fl_Align(FL_ALIGN_WRAP));
141         } // Fl_Button* p
142         { Fl_Output* o = pl = new Fl_Output(320, 120, 70, 25);
143             o->value(std::to_string(m.get_power()).c_str());
144         } // Fl_Output* pl
145         { r = new Fl_Button(320, 180, 70, 20, "Reset");
146             r->box(FL_ROUNDED_BOX);
147             r->color((Fl_Color)202);
148             r->labelcolor(FL_BACKGROUND2_COLOR);
149             r->callback((Fl_Callback*)cb_r);
150         } // Fl_Button* r
151         w->end();
152     } // Fl_Double_Window* w
153     w->show(argc, argv);
154     return Fl::run();

```

155 }

4.2 Lab9c.h

```
1 // generated by Fast Light User Interface Designer (fluid) version 1.0305
2
3 #ifndef lab9c_h
4 #define lab9c_h
5 #include <FL/Fl.H>
6 #include <iostream>
7 #include <string>
8
9 class Microwave {
10 public:
11     Microwave();
12     int get_time() const;
13     void increase_time();
14     void switch_power();
15     void reset();
16     void tick();
17     int get_power() const;
18 private:
19     /**
20      * stores time in seconds
21      */
22     int time;
23     int powerLevel;
24 };
25
26 Microwave m;
27 void cook_cb(void* op);
28 #include <FL/Fl_Double_Window.H>
29 extern Fl_Double_Window *w;
30 #include <FL/Fl_Box.H>
31 #include <FL/Fl_PNG_Image.H>
32 extern Fl_Box *b;
33 extern Fl_Box *c;
34 #include <FL/Fl_Output.H>
35 extern Fl_Output *op;
36 #include <FL/Fl_Button.H>
37 extern Fl_Button *s;
38 extern Fl_Button *a;
39 extern Fl_Button *p;
40 extern Fl_Output *pl;
41 extern Fl_Button *r;
42 #endif
```

5 Tex File

```
1 \input opmac
2 \input ../hisyntax
3 \tit Lab:Ch9c
4 \maketoc
5 Thu Pham
6
7 Profs. Topham
8
9 CS116
10
11 \filbreak
12 \sec Description
13 \begitems
14 * For this lab, we need to create a class Microwave into the GUI.
15 * We also implement four functions which are get time,
16   increase time, switch power, and reset.
17 * We want the user to control the microwave and let them
18   decide what they want.
19 * We need three buttons total in the GUI one for start,
20   one for add 30 sec, and the last one is for power switch.
21 * We want the output to display the minutes that remaining and begin with.
22 \sec The Problem
23
24 \picw=6in
25
26 \inspic lab9c.png
27
28 \filbreak
29 \sec The Result
30 This is the picture of the microwave
31 \picw=4in
32 \inspic micro.png
33
34 This is the picture of the cake
35 \picw=4in
36 \inspic cake.png
37
38 \filbreak
39 \sec Source Code
40 \filbreak
41 \secc Lab9c.cxx
42 \hisyntax{C}
43 \verbininput (-) lab9c.cxx
44
45 \filbreak
46 \secc Lab9c.h
47 \hisyntax{C}
48 \verbininput (-) lab9c.h
49
50 \filbreak
51 \sec Tex File
52 \hisyntax{C}
```



```
53 \verbatiminput (-) lab9c.tex
54
55 \bye
```