

Homework IV

Assigned in Laboratory V

Due Start of Laboratory VII

Please perform the following activities. This assignment is unpledged. You may work with one other person. You are allowed to talk with others, but your work must be your own.

Objective

Much of software development is taking existing code and modifying it. The modifications may be to add a new feature to the software, fix some bug, improve performance, or just do things better. Because existing software is continually updated and modified, it is important that it be constructed so that it is easy to change. In this assignment, we update the tax graph program in several ways. This should help you appreciate the difficulties of modifying existing code.

Problem

You are to modify your tax graph program. The modifications fall into two areas. First, the program is not structured very well. This was not your fault, because when you wrote the original version, you did not know about functions. Your first task is to restructure the program so that it is broken into smaller, more manageable pieces (i.e., functions). For example, it is common practice to have the input and output of the program handled by separate functions. One reason for this is that these sections of a program are the most likely to change. Therefore, it is nice to have these sections encapsulated or isolated in functions. As one part of this assignment, you should restructure your program to use functions where appropriate. At a minimum, the input and the output of the program should be handled by separate functions. There are other tasks or subtasks that could and should be handled by separate functions.

Before modifying any code, it would be a good idea to sit down and break the program up into logical tasks. Once you've done this, you might show your proposed design to a teaching assistant for feedback. We recommend that you do this before modifying your code. Do not wait till the last minute to get feedback!

The second area of modification concerns the display of the bar graph. As currently displayed, the graph is hard to interpret. Most bar graphs of this type use a stack of segments. When displayed this way, it is easy to see what percentage of the whole each tax bracket is contributing. Additionally, the current graph is hard to interpret because there is no legend. It would be nice to label the graph so that what each portion represents is clear. The EzWindows API provides a class that can help us do this. Look at page 736 in your book at the class `Label`. We can use this class to place text at appropriate points in the window. In particular, you can use it to label the segments of the bar graph. To use the class `Label`, you will need to place the following include statement in your program.

```
#include "label.h"
```

This statement must precede the include for `rect.h`.

Notes

The textual input and output of your program should be just like the original program. The following two screen captures demonstrate this aspect of your program.

```

H:\101\96\hw\hw03\taxgraph.exe
Enter the information for each tax bracket
For the first three brackets enter two numbers
separated by a space
The first number is the upper level of the bracket
The second number is the tax rate
A sample input might be
    15000 .02
which says the upper level of the bracket is $15,000,
and the tax rate for this bracket is 2%
For the last bracket, only enter the rate

Enter the values for Tax Bracket 1: 14999 0.02
Enter the values for Tax Bracket 2: 24999 0.03
Enter the values for Tax Bracket 3: 34999 0.05
Enter the tax rate for the income above $34999: 0.06
Enter the annual income: 56000

```

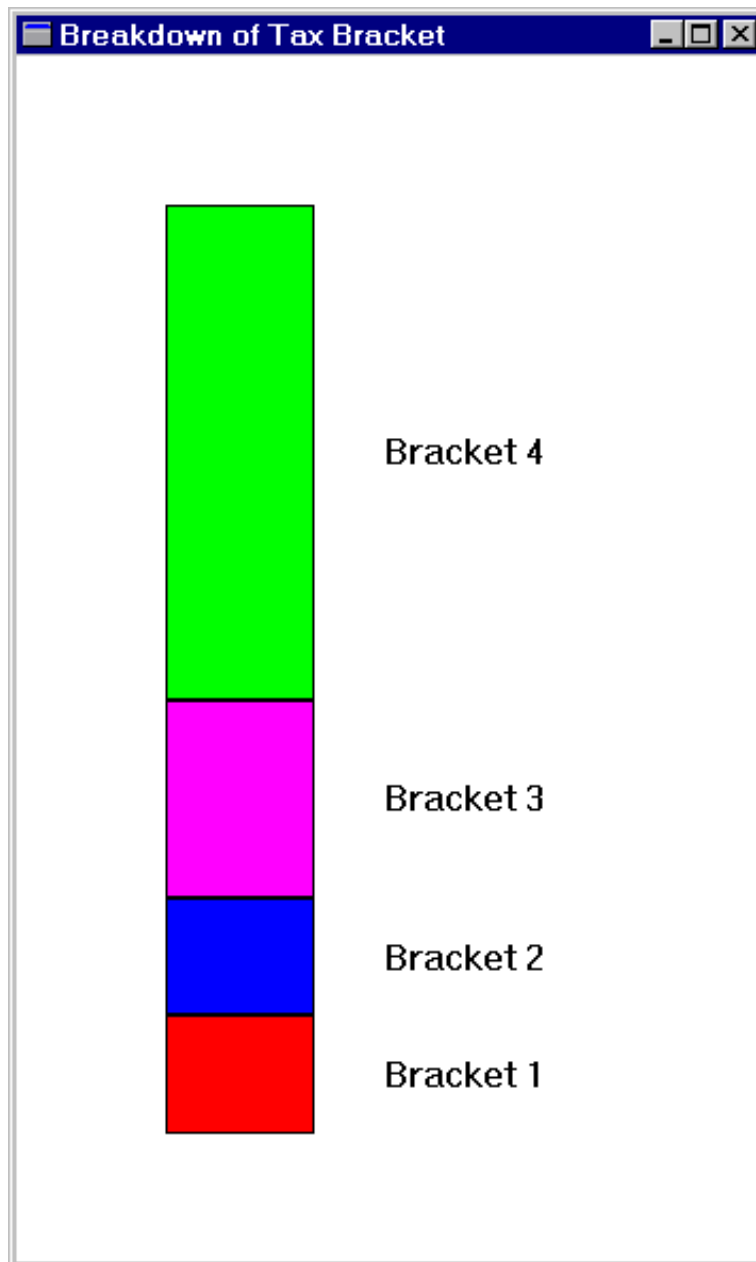
```

H:\101\96\hw\hw04\taxgraph.exe
Tax Table
Bracket  Income Range      Tax Rate
1         $0-$14999                0.02
2         $15000-$24999           0.03
3         $25000-$34999           0.05
4         Over $34999           0.06

Tax computation for an income of $56000 is:
Bracket 1 Tax =           $299.98
Bracket 2 Tax =           $300
Bracket 3 Tax =           $500
Bracket 4 Tax =          $1260
Total Tax =          $2359.98

```

In addition to this textual output, your program should use the EzWindows API to produce a bar chart illustrating the relative percentage of the total tax that is due to each bracket. The new chart for the above run looks like the following.



The height of your display window should be 16 centimeters. The width of your display window should be 10 centimeters. Tax bracket 1 should be displayed with a red bar; Tax bracket 2 should use blue; Bracket 3 magenta, and bracket 4 is green. The bottom of your bars should be approximately 1.5 centimeters from the bottom of the screen. The top of a bar should not be closer than 2.0 centimeters to the top edge of the window.

At this point, it would be good to clean out your submit directory. Do this by logging in to the CS1 Novell server and deleting any old files. After cleaning up your directory, you should submit

your program electronically before the beginning of Laboratory VII. See your handout for how to do this. You should call your file `hw04.cpp` when you copy it to the CS1 server. Do not copy the project file.

You should also turn in a hard copy (print out) at the beginning of your laboratory when the program is due.

See the 101 Web page for the grading criteria for this assignment. The URL is:

`http://www.cs.virginia.edu/cs101/`