Homework 3 - Ascii Diamonds

Due: Friday, November 6th, 2015

Visual Basic is so boring to look at. Everything is grey or blue and just lame overall. Let's change that by making some art!

Using your amazing Visual Basic skills, you will make a program that displays some pretty ASCII art, which is drawing shapes with letter, symbols, and whitespace. To do this you will need to use at least one loop.

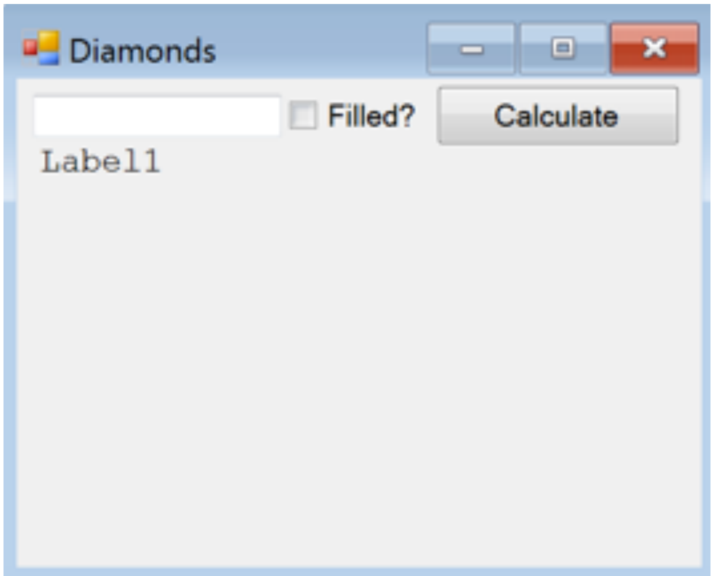
Your program should take one number and a checkbox as the input. This number will be the length of the sides of the art. For example, if someone input a 5 and checked the "filled" checkbox your program should display.

```
  *
 ***
*****
 ***
  *
```

If the filled is not checked:

```
  *
 * *
*   *
 * *
  *
```

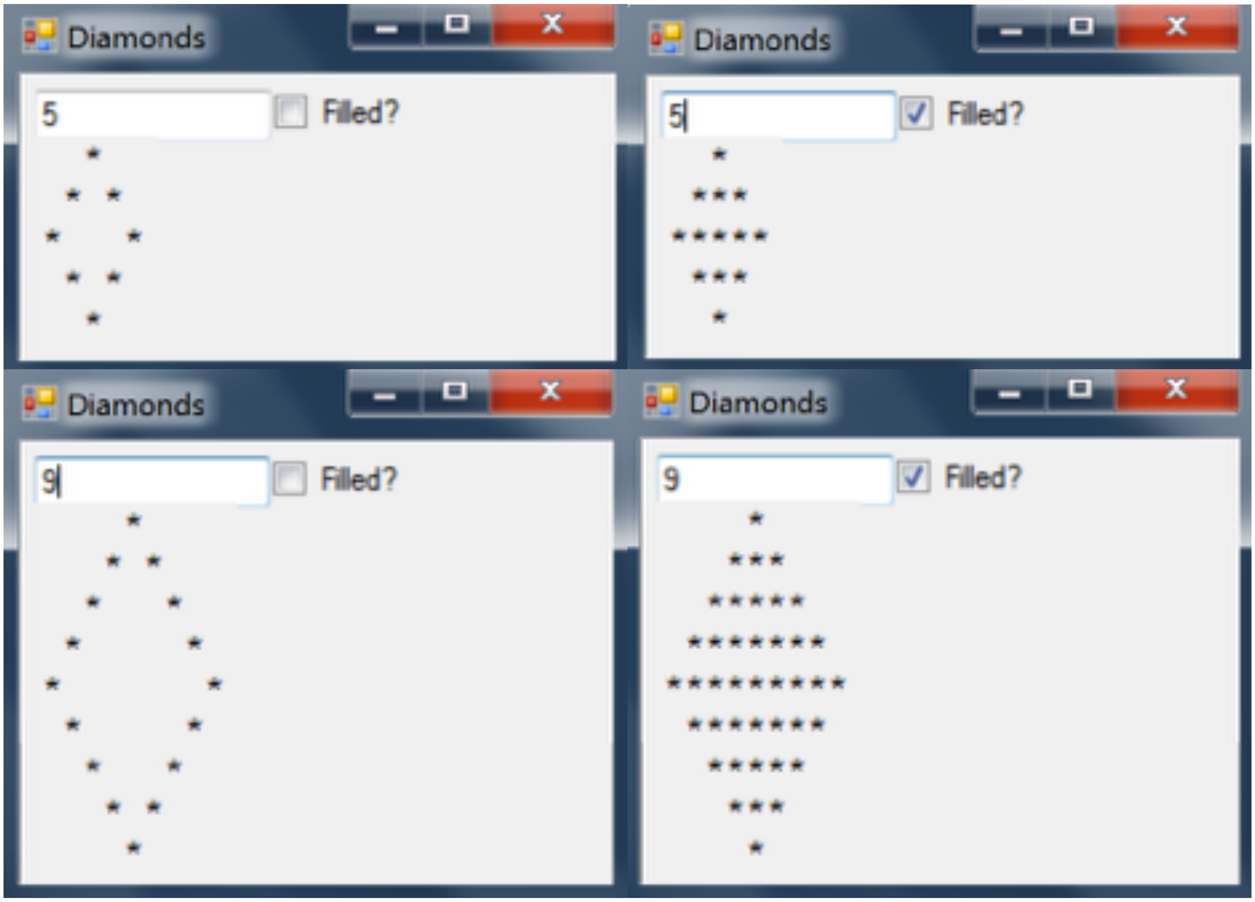
Recreate this GUI.



More examples on the next page.

P.S. You can use any symbol you like for the "*" as long as it produces an easily recognizable diamond of the correct size.

Which when you are done will create something like this:



5 □ Filled?

```
  *
 * *
*   *
 * *
  *
```

5 ☑ Filled?

```
  *
 ***
*****
 ***
  *
```

9 □ Filled?

```
    *
   * *
  *   *
 *     *
*       *
 *     *
  *   *
   * *
    *
```

9 ☑ Filled?

```
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

Things of note:

- There is only one label.
- In order to use only one label like this you will need to “concatenate” strings together. This is like adding, but for strings. `1 + 2 = 3` but `“1” & “2” = “12”`
- At then end of each row concatenate on a vbNewLine to start the next row. Try this command: `Label1.Text = “First Line” & vbNewLine & “Second Line”`
- You **must** set the font on the ascii art label to a monospaced font such as `Courier New.` A monospaced font is one where every character is the same width so that they line up in columns nicely. Monospaced fonts are ubiquitous in programming, Visual Studio uses a monospace font.
- (When I wrote this I made it redraw the shape every key press. You don’t have to do this, you can use a button like in the first picture.)

Hints:

There are three ways to solve this problem

1. Add the correct number of spaces, then the “*”s then the same number of spaces again.
    1. This is hard to do, especially for the un-filled diamond.
2. Get the x,y value of each character and put the characters in based on purely their position.
    1. This is easier but requires you to account for many cases and write lots of conditional statements based on equations that you would have to invent.
3. Use a concept called the “Manhattan distance”/Taxicab geometry. I suggest you use this method. Filled and unfilled diamonds are almost identical with this concept.
    1. In taxicab geometry, you are only able to count distance along the x and y axis. Picture that you are driving on the streets of Manhattan; to go between two street corners you can’t just drive through the buildings (using traditional distance formula). Instead you must drive along the streets. This concept can be used to easily decide whether a certain square should be marked as a “*” or a “ ”.

To calculate the Manhattan distance between two points use:

$$d(p_{souce}\ ,\ p_{destination}) = |x_{destination} - x_{source}| + |y_{destination} - y_{source}|$$