Lab 4: Pong

There is a basic version that everyone must do, followed by several options. If you choose to do the options, they MUST be done in order.

When you get the basic version working, copy your file and work on option 1 in the new file. Once you get option 1 working, copy that file and work on option 2 in the new file, etc. This way you won't lose your working versions. When done, submit only your most recent working file. DO NOT SUBMIT partially working code. I won't be able to tell what works and what doesn't, so I'll have to assume the entire program doesn't work. When you submit your lab to PowerSchool, please let me know which options, if any, you've implemented.

In order to receive credit, your code MUST BE WELL COMMENTED. This includes (a) commenting most lines of code, and (b) putting a comment at the beginning of each section of the code.

Example:

```
// Set the screen to black
@SCREEN   //A=address of screen
M=-1      // Set the pixel to black
…

// Set the screen to white
…

// Determine if a key is pressed
```

Basic version (must be done first): Paddle starts and stays on the bottom row and moves left and right via the left and right arrows. Paddle has a height of 3 rows and a width of 3 words. It stops moving when either (a) the arrow key is released, or (b) when it reaches the edge of the screen. For the basic version, the paddle is allowed to move in word (i.e. 16-bit) increments. *Add a delay loop so the paddle movement can be seen.*

Option 1: When the down arrow is pressed, a small black square (the ball) drops from the top of the screen straight down. The down arrow is pressed just long enough for the program to recognize it, and it can then be released. The ball starts in any column you want. The ball should move at a reasonable speed so you can easily see it drop. For this option the box simply drops

until it reaches the bottom of the screen (even if it collides with the paddle) and disappears, after which the game is over. Only one ball is in play at a time. For full credit, the paddle should be able to move while the ball is dropping.

Option 2: If the ball touches any part of the paddle, the ball reverses direction and goes straight up until it reaches the top of the screen and disappears. If the ball hits the bottom of the screen, it disappears and the game is over as before.

Option 3: Same as Option 3 except if the ball hits the top of the screen it reverses direction and goes straight down.

Option 4: Smoother paddle movement – move the paddle left and right one pixel at a time (vs. one word at a time in the basic version). For full credit include a short write up describing your algorithm for moving by one bit.

Grading: I am looking for working code, simplicity, clarity, and sensible documentation (COMMENTS!). Make sure you submit a write up if you choose to complete option 4. **Note that options 3 and 4 are bonus options!**

Program	**Working?**	**Well-built?**	**Comments**
Basic Pong	/ 56	/ 29	
Option 1	/ 6	/ 4	
Option 2	/ 3	/ 2	
Option 3	/ 5		
Option 4	/ 5		
Total	/ 65	/ 35	

Implementation Strategy

For the basic version, implement each step BEFORE continuing

1. Draw the paddle
2. Move the paddle ONE word to the right (no key pressed)
3. Move the paddle ONE word to the left (no key pressed)
4. Move the paddle ONE word to the right (then stop) if the right arrow is pressed
5. Move the paddle ONE word to the left (then stop) if the left arrow is pressed
6. Move the paddle left and right as long as an arrow key is pressed
7. Stop moving the panel if it reaches the end of the screen

For each option, write out the steps you will follow (as above) BEFORE starting to code