Practicum Rules

- This is a multi section exam that will be given to different midshipmen at different times. As per USNAINST 1531.53A, you may NOT communicate about this exam with anyone using any medium until your instructor tells you that you can.
- You may use your written notes, your own code stored on your CS Department home directory, the course website, and online resources that are linked directly from the course website.
- You may not communicate with anyone other than your instructor while taking this exam.
- You may submit as many times as you like, your most recently submitted solution for a given part will be the one graded.
- Your most correct submitted solution will be graded out of 60 points. Your second most correct will be graded out of 25 points, your third most correct out of 15 points. So focus first on getting one problem completely correct. Programs should function correctly. Partial credit for incorrect solutions will be extremely limited.
- **Your program's input/output behavior must match the examples exactly!**

How to submit a solution

You will be using the submit script to submit your practicum solutions.

1. To submit your first solution give the command:

   ```
   submit p1pr12wk p1pr12wk.cpp
   ```

2. To submit your second solution give the command:

   ```
   submit p2pr12wk p2pr12wk.cpp
   ```

3. To submit your third solution give the command:

   ```
   submit p3pr12wk p3pr12wk.cpp
   ```

Problem 1

Write a program named p1pr12wk.cpp that

1. reads in a sequence of integers (how many appears at the start of the input),
2. reports how many zeros, negative and positive values there were, and
3. prints out the positive values from the input in order.

```
~/$ ./p1
10 : 2 -1 3 0 -9 0 3 8 7 -4
#zeros = 2, #negatives = 3, #positives = 5
Positives = 2 3 3 8 7
```

```
~/$ ./p1
15 : 23 -9 -3 -2 14 2 5 5 18 -4 7 5 6 -1 1
#zeros = 0, #negatives = 5, #positives = 10
Positives = 23 14 2 5 5 18 7 5 6 1
```

Problem 2

Write a program named p2pr12wk.cpp that decodes a secret message hidden *steganographically* in the following way:

1. The user enters a number *n* followed by *n* "keywords", each space separated, then the user enters the name of a file that contains a hidden message. The file contains some number of space-separated words, then ends with a final space and then a "." character. If the file doesn't exist your program must print out an error message (as shown below) and exit.

2. The hidden message consists of every word in the text that immediately follows a keyword, so if the keywords are "please" and "shark" the text below yields the message "don't panic"

```
we ask you please don't feed my shark panic may ensue .
                    |                   |
                    |                   |
                  don't               panic
```

3. If we have two keywords in a row, the second doesn't act as a keyword, instead it is part of the message: e.g. with keywords "so", "rain" and "in" we get

```
the rain in spain falls mainly in the plain or so eye ear .
          |                       |              |
          |                       |              |
         in                      the            eye
```

Sonme sample files are msg1.txt, msg2.txt, msg3.txt, and msg4.txt.

```
~/$ ./p2
2 are a
foobar.txt
File 'foobar.txt' not found!
```

```
~/$ cat msg2.txt
there are good resons to
refuse a job at subway .
~/$ ./p2
2 are a
msg2.txt
good job
```

```
~/$ cat msg3.txt
my only real regret is that i didn't
make a better practicum  ...  that
was probably a bit long and i fear
that great pain may have been caused .
~/$ ./p2
3 is that better
msg3.txt
that practicum was great
```

```
~/$ cat msg1.txt
If you see me then meet someone else
who me and you think seems at least
kind of silly then noon or sometime
thereafter is probably too late .
~/$ ./p2
3 who seems then
msg1.txt
meet me at noon
```

```
~/$ cat msg4.txt
look over there at the tree ; it is pink I had no idea such things
exist ; if you try to envision one you'll fail because there isn't
anything quite like such a tree ; it is small because only small
plants avoid being cut down ; you do not have to take my word for this
since believe it or not we live near one ; at least you do , maybe I
do not ; who knows .
~/$ ./p2
6 over it had you because do
msg4.txt
there is no try there is only do or do not
```

Problem 3

Write a program named p3pr12wk.cpp that provides simple graphics functionality. The user will specify a grid size (numrows x numcols) and then use a simple command language:

- `quit` - which quits the program, obviously
- `show` - which shows the current drawing
- `draw` - which has a number of options
  - `draw @ i j` - draws at row *i*, column *j*. note: @ always draws with a '+' character, overwriting whatever else is there.
  - `draw row i` - draws across all positions in row *i*. note: row draws with '-' characters, but if a square already has a '|' or a '+' character, it will draw a '+' rather than '-'.
  - `draw col j` - draws across all positions in column *j*. note: col draws with '|' characters, but if a square already has a '-' or a '+' character, it will draw a '+' rather than '|'.

Note: Try in3a.txt as shown in the third example below. The shell command `./p3 < in3a.txt` runs program p3 sending it the contents of file in3a.txt as the input to standard in, as if that's what the user had typed.

```
~/$ ./p3
5 x 10
draw row 2
draw col 4
show
    |
    |
----+-----
    |
    |
draw @ 3 7
show
    |
    |
----+-----
    |  +
    |
quit
```

```
~/$ ./p3
6 x 9
draw row 2
draw col 7
draw @ 0 8
draw @ 5 0
show
       |+
       |
-------+-
       |
       |
+      |
quit
```

```
~/$ ./p3
5 x 10
draw row 0
draw row 4
draw col 0
draw col 4
show
+---+-----
|   |
|   |
|   |
+---+-----
quit
```

```
~/$ ./p3
5 x 20
draw row 0 draw row 4
draw col 0 draw col 19
draw row 2 draw col 10
draw @ 1 1 draw @ 3 5
show
+---------+--------+
|+        |        |
+---------+--------+
|    +    |        |
+---------+--------+
quit
```

```
~/$ ./p3 < in3a.txt
+---+---+---+---+---+
| + |   | + |   | + |
+---+---+---+---+---+
|   | + |   | + |   |
+---+---+---+---+---+
| + |   | + |   | + |
+---+---+---+---+---+
|   | + |   | + |   |
+---+---+---+---+---+
| + |   | + |   | + |
+---+---+---+---+---+
```