(48024) Applications Programming Patterns Lab tips and suggestions

This lab is all about building a program by connecting patterns together. Knowing the patterns is only part of the story. The other part is being able to combine them together in a logically consistent way. This document aims to provide a few helpful tips and suggestions.

SUM, COUNT, MIN, MAX have two parts

Each of these patterns have two parts. The first part goes above the loop, while the second part goes inside the loop. For example, here is the count pattern:

```
int count = 0; // first part above the loop
<for each value> {
    count++; // second part inside the loop
}
```

And here's the max pattern:

```
int max = Integer.MIN_VALUE; // first part above the loop
<for each value> {
    if (value > max) // second part inside the loop
        max = value;
}
```

For the moment, what is important is just to notice these two parts. Read more below.

Merge multiple patterns into a single loop

If you need to use more than one pattern in a loop, don't create a separate loop for each pattern. Instead, you should merge multiple patterns into a single loop.

e.g.

Specification: Read integers until the user enters -1. Show the biggest and smallest number.
Solution: We merge the min/max patterns into a single read loop, then output the result.

Start with the read loop framework:

```
System.out.print("Value: ");
```

```
int value = In.nextInt();
while (value != -1) {  // READ LOOP pattern
    <<Use the value>>

    System.out.print("Value: ");
    value = In.nextInt();
}

<<Output the results here>>>
```

That's a good start. Now to merge in the min and max patterns, remember that part goes above the loop, and part goes inside the loop. For the part that goes inside the loop, we will place it at the position where it says <<Use the value>>. For the part that goes above the loop, you can place this right at the top. The finished program is:

```
// Part 1 of min goes ABOVE the loop
int min = Integer.MAX_VALUE;  // MIN pattern (part 1)
// Part 1 of max goes ABOVE the loop
int max = Integer.MIN_VALUE;  // MAX pattern (part 1)

System.out.print("Value: ");
int value = In.nextInt();
while (value != -1) {
    // Part 2 of min goes INSIDE the loop
    if (value < min)  // MIN pattern (part 2)
        min = value;
    // Part 2 of max goes INSIDE the loop
    if (value > max)  // MAX pattern (part 2)
        max = value;

    System.out.print("Value: ");
    value = In.nextInt();
}

System.out.println("Min value = " + min);
System.out.println("Max value = " + max);
```