How to Write a Program

We are to write a program that reads a line of input. If the line contains capital letters, the program will print "Capitals". If it doesn't contain any capital letters, the program ill print "No capitals". This handout shows you how to do this.

First, we have to write down a detailed set of steps that will carry out this action. This set of steps is called the *algorithm*.

How To Do It

The first thing to do is read in a line of input. If the user typed something, the program looks at each character on the line, starting with the first, to see if it is a capital letter. If it is, the program prints "Capitals" and stop at once. If not, the program goes on to the next character, and repeats this process until the program has checked all the characters in the line. If the program runs out of characters in the line, it prints "No capitals".

If the user didnt type anything, though, the program just prints "No capitals" at once.

Here is the above, written down step by step.

```
1   capital letters are ABCDEFGHIJKLMNOPQRSTUVWXYZ
2   read in a line
3   is there a character on the line?
4   if yes
5     set the current character to it
6     while there's more input and the current character is not a capital letter
7       advance to the next character
8     if there's more input
9       print "Capitals"
10    otherwise
11      print "No capitals"
12  otherwise
13    print "No capitals"
```

First Version

This is a direct translation of the above into Python. I put the numbers of the lines in the step-by-step instructions into comments at the right of each Python line. If you want to run this, you have to get rid of those numbers!

This program is on the class web site, as "anycaps1.py".

```
1   CAPITALS="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
2   line = input("What's your line? ");
3   length = len(line);
4   if (length > 0):
5       index = 0;
6       while (index < length) and not (line[index] in CAPITALS):
7           index = index + 1
8       if index < length:
9           print "Capitals"
10      else:
11          print "No capitals"
12  else:
13      print "No capitals"
```

How To Do It, Shorter

Now let's see if we can shorten the previous algorithm a bit. In it, we determine whether we saw a capital letter by whether we looked at every character in the line (look at line 8, and see how we decide whether to print "Capitals" or "No capitals". That doesn't really reflect what we want (although it does work); we want to print based on whether there is a capital. So let's focus on doing that.

We'll define a variable that says whether or not we've seen a capital letter. Initially, of course, we've not seen one. In the loop in lines 5–6, we'll check the current letter to see if it's a capital letter. If it is, we'll change the value of the variable to indicate we have seen a capital letter. Then, when we reach the end of the loop, we check whether we have seen a capital letter. If so, we just drop out of the loop. If not, and there are more letters in the line, we continue through the loop again.

Then, at the end, we just check the variable to see if we saw a capital letter. If we did, we print "Capitals"; if not, we print "No capitals".

Here are the modified, step-by-step instructions. In this, "←" is assignment.

```
1   capital letters are ABCDEFGHIJKLMNOPQRSTUVWXYZ
2   read in a line
3   seencapital ← "no"
4   start at first character
5   while seencapital is "no" and there is a character remaining on the line
6     if current character is a capital
7       seencapital ← "yes"
8     advance to the next character
9   if seencapital is "yes"
10    print "Capitals"
11  otherwise
12    print "No capitals"
```

Second Version

This one, again, is a line-by-line translation into Python. It's also up on the class web site, as "anycaps2.py".

```
1   CAPITALS = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
2   line = input("What's your line? ");
3   seen_capitals = "no"
4   index = 0
5   while seen_capitals == "no" and index < len(line):
6       if line[i] in CAPITALS:
7           seen_capitals = "yes"
8       index = index + 1
9   if seen_capitals == "yes":
10      print "Capitals"
11  else:
12      print "No capitals"
```

Third Version

This program is similar to the previous one, but has a few idiomatic differences. In Python, we can use a "for" loop to step through a string. This means we don't have to test for the length of the string, or keep track of the current character; the "for" loop does it for us. This means we can eliminate lines 4 and 8. But it also means we must change the condition in line 5, because there's no way to tell a "for" loop to stop in the middle of the string.

So we don't try. Instead, we arrange that the program will drop out of the "for" loop when it sees the first capital letter. We do this using a "break" statement. That statement says, essentially, that the next statement to do is the first statement after the "for" loop.

So, here's the revised program. It does exactly what the previous program does, but it's a bit less cluttered—and, I think, may be easier to read. As with the other programs, this is also on the class web site, as "anycaps3.py".

```
 1  CAPITALS = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
 2  line = input("What's your line? ");
 3  seen_capitals = "no"
 4  for i in line:
 5      if i in CAPITALS:
 6          seen_capitals = "yes"
 7          break
 8  if seen_capitals == "yes":
 9      print "Capitals"
10  else:
11      print "No capitals"
```