Knitters and coders: separated at birth?

People often say that computers are all around us, but you could still escape your phone and iPod and go out to the park, far away from the nearest circuit board if you wanted to. It’s a lot more difficult to get away from the clutches of computation itself though. For one thing, you’d have to leave your clothes at home. Queen Mary Electronic Engineer Karen Shoop tells us about the code hidden in knitting, and what might happen when computers learn to read it.

If you’re wearing something knitted look closely at it (if it’s a sunny day then put this article away till it gets colder). Notice how the two sides don’t look the same: some parts look like a raised ‘v’ and others like a wave pattern. These are made by the two knitting stitches: knit and purl. With knit you stick the needle through and then behind the knitting; with purl you stick the needle in the other direction, starting behind the knitting and then pointing at the knitter (see picture). Expert knitters know that there’s more to knitting than just these two stitches, but we’ll stick to knit and purl. As these stitches are combined, the wool is transformed from a series of waves or ‘v’ into a range of patterns: stretchy stripes (ribs), raised speckles (moss), knots and ropes (cable). It all depends on the number of purls and knits, how they are placed next to each other and how often things are repeated.

A rib would look like:

‘5k, 5p, then repeat this [a certain number of times], then repeat the line [another number of times ]’

To a computer scientist or electronic engineer all this looks rather like computer code or, to be precise, like the way of describing a pattern as a computer program.

How your jumper is like coding

So look again at your knitted hat/jumper/cardigan and follow the pattern, seeing how it changes horizontally and vertically. Just as knitters give instructions for this in their knitting pattern, coders do the same when writing computer programs. Specifically programmers use things called regular expressions. They are just a standard way to describe patterns. For example a regular expression might be used to describe what an email address should look like (specifying rules such as that it has one ‘@’ character in the middle of other characters, no full-stops/periods immediately before the @ and so on), what a phone number looks like (digits/numbers, no letters, possibly brackets or hyphens) and now what a knitting pattern looks like (lots of k’s and p’s). Regular expressions use a special notation to precisely describe what must be included, what might possibly be included, what cannot be, and how many times things should be repeated. If you were going to teach a computer how to read knitting patterns, a regular expression would be just what you need.

Knitting a regular expression

Let’s look at how to write a knitting pattern as a regular expression. Let’s take moss or seed stitch as our example. It repeats a “knit one purl one” pattern for one line. The next line then repeats a “purl one knit one” pattern, so that every knit stitch has a purl beneath it and vice versa. These two lines are repeated for as long as is necessary. How might we write that both concisely and precisely so there is no room for doubt?

In knitting notation (assuming an even number of stitches) it looks like:

Row 1: ‘k1, p1; rep from *’
Row 2: ‘p1, k1; rep from *’
or
Row 1: (K1, P1) rep to end
Row 2: (P1, K1) rep to end
Repeat these 2 rows for length desired.

All this is fine ... if it’s being read by a human, but to write experimental knitting software the knitting notation we have to use a notation a computer can easily follow: regular expressions fit the bill. Computers do not understand the words we used in our explanation above: words like ‘row’, ‘repeat’, ‘rep’, ‘to’, ‘from’, ‘end’, ‘length’ and ‘desired’, for example. We could either write a program that makes sense of what it all means for the computer, or we could just write knitting patterns for computers in a language they can already do something with: regular expressions. If we wanted to convert from human knitting patterns to regular expressions we would then write a program called a compiler to do the translation.
In a regular expression, to give a series of actions, we just name them. So kp is the regular expression for one knit stitch followed immediately by one purl. The knitting pattern would then say repeat or rep. In a regular expression we group actions that need to be repeated inside curved brackets, resulting in (kp). To say how many times we need to repeat, curly brackets are used, so kp repeated 10 times looks like this: (kp){10}.

Since the word ‘row’ is not a standard coding word we then use a special character, written \n, to indicate that a new line (=row) has to start. The full regular expression for the row is then (kp){10}\n. Since the first line was made of repetitions of kp the following line must be made of repetitions of pk, or (pk){10}\n.

These two lines have to be repeated a certain number of times themselves, say 20, so they are in turn wrapped up in yet more brackets, producing: ((kp){10}\n(pk){10}\n){20}.

If we want to provide a more general pattern, not fixing the number of kp in a row or the number of rows, the 10 and 20 can be replaced with what are called variables - \(x\) and \(y\). They can each stand for any number, so the final regular expression is: ((kp){x}\n(pk){x}\n){y}.

How would you describe a rib as a regular expression (remember, that’s the pattern that looks like stretchy stripes)? The regular expression would be ((kp)(x)\n(y)).

Regular expressions end up saying exactly the same thing as the standard knitting patterns, but more precisely so that they cannot be misunderstood. Describing knitting patterns in computer code is only the start, though. We can use this to write code that makes new patterns, to find established ones or to alter patterns, like you’d need to do if you were using thicker wool, for example. An undergraduate student at Queen Mary, Hailun Li, who likes knitting, used her knowledge to write an experimental knitting application that lets users enter their own combination of ps and ks and find out what their pattern looks like. She took her hobby and saw how it related to computing.

Look at your woolly jumper again...it’s really made out of computation!