Problem Solving and Prior Knowledge

by **Colin Foster**

David Ausubel (1968, p. *iv*) famously said that, "If I had to reduce all of educational psychology to just one principle, I would say this: The most important single factor influencing learning is what the learner already knows; ascertain this and teach [them] accordingly". Effective teaching of any mathematical content has to take account of the learner's prior knowledge and build up from there. But, what does this mean for the teaching of *problem solving*? If problem solving is defined as "engaging in a task for which the solution method is not known in advance" (NCTM, 2000, p. 52), then what would it mean to build on what the learner already knows? Is the role of prior knowledge in problem solving a purely *negative* one? The teacher's job is to try to ensure that the student does *not* know anything relevant to the problem, so that the problem is truly 'problematic' for the student, and not merely a routine application of something that they have been taught (Foster, 2019). So the less prior knowledge the better?

It's certainly true that prior knowledge can inhibit problem solving. When the physicist Richard Feynman was asked what he might wish to have done differently in his career, he replied that he "would try to forget how I had solved a problem. Then, each time the problem arose, I might solve it in a different way – I wouldn't be thinking about how I had solved it before" (Feynman, Feynman, & Ferris, 2006, p. 370). This resonates with the maxim attributed to George Pólya that, "It is better to solve one problem five different ways, than to solve five problems one way", and highlights how difficult it can be to 'unsee' what we have previously seen. We will all have been in mathematics professional development sessions where the person leading it suggests we work on a mathematical problem, and it turns out to be one that, by luck, we happen to know quite well. Maybe it's a favourite problem that we often use with students, and feel that we know 'inside out'. Our task then becomes avoiding spoiling the problem for anyone else, and seeking out the hidden depths, which any good problem will have, by extending it, or solving it a different way. However, our knowledge of the problem is preventing us coming at it with fresh eyes. If problem solving is about dealing with *unfamiliar* situations, where you *don't* know what to do, then, if you happen to be aware of a ready-made method, you don't get the chance to do that. The satisfaction from being able to get 'the answer' efficiently is often in tension with the satisfaction from being able to puzzle something out.

As it says on the T-shirt, "Mathematicians aren't the people who find maths easy. They're the people who enjoy how hard it is". To that end, sometimes we might even choose to deliberately *make things harder* for ourselves. For example, when tackling a geometry problem, we might decide *not* to allow ourselves to use dynamic geometry, at least until we have formed some pretty strong conjectures that we want to test out. It is not so much that using dynamic geometry would be 'cheating' as that it would spoil the process for us, and we would consequently get less out of it (i.e., 'cheating ourselves'). Dynamic geometry can be a great tool for developing visual imagery, but, if you're not careful, it can also be a great tool for *avoiding* having to use visual imagery, and so, if overused, might leave those skills underdeveloped. Another example when solving a geometrical problem could be deciding that you want to do it 'classically', without, say, using any trigonometry or coordinate geometry. Or you might want to tackle a 'simultaneous equations' kind of problem only using logical argument, without employing any symbolic algebra. All of these can be ways of keeping the problem-solving aspects alive, despite relevant things that we might be able to access – almost pretending not to know things that we do know and deliberately constraining ourselves. Imagine having a machine that would solve all your mathematical problems for you – your first reaction might be that this sounds great! But, if you enjoy problem solving, then having a machine that would solve all your problems for you would be rather like having a machine that would read all your books for you, or a machine that would eat your ice cream for you – no thanks!

So, being able to engage in problem solving depends to some degree on a *lack* of prior knowledge. But, on the other hand, the ideal problem solver cannot simply be an empty vessel – the more ignorant the better! There must be something that the problem solver brings to the problem besides nebulous qualities like creativity, ingenuity and flair. Unless they also come with a toolbox full of relevant techniques, then they will surely fail to make progress with any problem. Indeed, unless they have previously met a problem at least a little bit like the one they are tackling, it is hard to see how they could possibly get anywhere with it. The expert problem solver has a wide range of tools at their disposal, honed through extensive use, combined with experience in wielding the right one at the right time. So, does this mean that problem solving is nothing more than the ability to select a suitable method from among a large number of possible methods, and then carry out that process accurately? Surely we don't expect our students to creatively manufacture a brand-new, never-before-seen method (at least not by them) all by themselves, from nowhere?

I got thinking about all of this while watching a problem-solving lesson in which the teacher began by asking the students what the first thing is they should do when they are given a mathematical problem to solve. The desired answer was: "Ask yourself whether you've seen anything like it before." For me, this gets to the heart of this tension over teaching problem solving. If you've seen something very like this problem before, then for you this isn't problem solving – it's following a known method. Of course, that is great if you just want the solution to the problem – in real-life situations, we would be very pleased. Problem solving is slow, hard work, and in the real world we would only do it when a ready-made method is not available. However, it's different if your purposes are *educational*. If you can reach for a known method off the shelf that will solve your problem, then that is simply the normal intended experience of learning school mathematics. The 'problem' has turned into an exercise in applying a standard method. The only 'problematic' feature might be *finding* this method among all the other ones you have been taught, or maybe deciding which of several possible methods will be most convenient or efficient in this particular case. And this is just exactly the pattern of normal (non-problem-solving) school mathematics lessons. The teacher tells you about a known, supposedly important problem – and then (usually in the next breath, without any time to 'stew') shows you a general method of solving it [see Note]. Although this was undoubtedly real problem solving for whoever first came up with the method, perhaps hundreds of years ago, for those of us born too late for that, it's just imitating a standard procedure, not problem solving.

One way to think about how mathematics develops is to see it as having problems at the heart (Halmos, 1980). Over time, we gradually notice that certain kinds of problems keep coming up, and different problems have similarities, and so we organise the problems that we meet into categories and then devise general methods for solving them. This is great, because it means that we don't have to start from scratch with every problem. For example, we can use 'rules' to differentiate, like the 'product rule', rather than having to differentiate from first principles every time (see Phillips, 1938). If a class of problem turns up fairly often, and is important enough for us to want efficient solutions, then it is worth formulating a general method of solution – a recipe for getting the answer. Of course, this kills 'the problem' by replacing it with a procedure – and that is the whole point. The more methods we have available, the more powerful our toolbox has become. The story of mathematics can be seen as the story of replacing problems with methods, and growing our toolboxes. But, then how do you avoid also replacing mathematical problem solvers with technicians merely following rules?

I remember being puzzled by this at university. I found 'differential equations' a totally 'cookbook' topic. The approach seemed to be: "If ever you happen to stumble across a differential equation of *this* form, then here's a nice approach you can take to solve it." That was all very well, but I wanted to say, "But what if that x^2 were an x^3 instead?" The answer would have been, "Oh, well of course this method wouldn't work then. And there's probably no analytical method that would – you'd have to do it numerically." At the time, this left me feeling that differential equations wasn't a proper subject! I was just being asked to learn a lot of special cases, and I could only use those techniques in those particular, very narrowly-defined circumstances. Of course, those methods had been developed because those circumstances came up a lot in applications, and they were undoubtedly clever strategies, but my feeling was that all of this was useful, but not very satisfying. Pólya quipped, "In order to solve this differential equation you look at it till a solution occurs to you." But what is going on here – just comparing with all the known types of differential equation in your long-term memory, looking for a match? It surely does not mean inventing a new method every time.

Returning to our teacher and his problem-solving lesson, maybe there's a different interpretation of what he is doing here. He is perhaps not asking the students to consider whether they have a ready-made *method* for solving this particular problem, but whether any general problem-solving *strategy or approach* that they have used before might be relevant for solving this one? Any teacher will have seen students who, in the right circumstances, would be able to recall and accurately use a certain method, but who do not think of that method at the appropriate time when faced with a problem for which it would be ideal. This happens to me all the time too. There are several fairly generic problem-solving strategies that can help in those situations. For example, for a geometrical problem, if you see nothing useful in the diagram, you sometimes find that by rotating the diagram you suddenly spot a helpful relationship that you hadn't noticed. So, 'If you don't see anything useful, try rotating the page' can become a productive heuristic or strategy. And, if these strategies are useful – and sometimes they are, in game-changing ways – perhaps they are what we should be teaching when we are 'teaching problem solving'?

In my experience, many 'problem-solving lessons' are little more than 'problem-solving opportunities'. The teacher provides a problem, and then the students try to solve it. Maybe some of them manage it, and others don't. With luck, there might be time for a brief plenary at the end of the lesson, in which a successful student (or the teacher, if necessary) tells everyone what they did. And then the lesson ends – that was 'problem solving' (Foster, 2019). This might be 'doing' problem solving, but it doesn't seem like it's 'teaching' problem solving unless students are, as a result, better able to solve future problems that they have not been specifically prepared for (see Sweller, 1988). Perhaps, instead, alongside the ever-growing toolbox of methods that students learn in school, we should be explicitly teaching *problem-solving strategies*. By this, I mean not just mentioning them in an opportunistic way, whenever a student happens to discover one while working on a problem, but planning to teach them deliberately and explicitly. Indeed, perhaps this should be the main criterion for how we select the problems that we present students with in a problem-solving lesson. If we want them to learn the 'If you don't see anything useful, try rotating the page' strategy, then we need to find a problem that is dramatically unlocked by using this strategy. Using this, we try to create a memorable experience where 'turning the page' was like a magical action. This needn't take a whole lesson – the point could perhaps be made in a 10-minute episode. Then, we carefully sequence a series of problems, each of which demonstrates the value of a particular general strategy, that we explicitly teach, and which also may provide opportunities for students to review and practise using previously-taught strategies. Then we can ask students which other problems they have encountered before that might be unlocked by this same strategy – and can they invent a new problem for which this strategy could be the key? This seems like a way that might help students become better and better problem solvers.

Sometimes, the impression is given that for a problem-solving lesson all you need is a problem of the right level of difficulty, which students won't have the prior knowledge to immediately know how to solve. That leaves us with an enormous range of possible problems to use with any group of students, and can make the choice of a problem seem rather haphazard. Instead, I think we need to narrow this field considerably by looking at some of the great classic school mathematics problems, auditing them for key strategies, and sequencing them into a learning trajectory focused on those strategies. That is the approach we are currently trying to adopt here at Loughborough in our work towards designing a problem-solving teaching programme for a school mathematics curriculum.

Note

Of course, when preparing for an examination, it's a bit harder, because you are faced with mixed questions, and have to select the right method for each one. But, hopefully, you have been taught a suitable method for every question, and it's merely a matter of selection.

References

Ausubel, D. (1968). *Educational psychology.* New York: Holt, Rinehart & Winston.

Feynman, R. P., Feynman, M., & Ferris, T. (2006). *Perfectly reasonable deviations from the beaten track: The letters of Richard P. Feynman*. London: Basic Books.

Foster, C. (2019). The fundamental problem with teaching problem solving. *Mathematics Teaching,* **265**, pp. 8–10.

Halmos, P. R. (1980). The heart of mathematics. *The American Mathematical Monthly,* **87** (7), pp. 519-524.

National Council of Teachers of Mathematics (NCTM) (2000). *Principles and standards for school mathematics*. Reston, VA: NCTM.

Phillips, E. G. (1938). On differentiation from first principles. *The Mathematical Gazette*, **22** (251), pp. 374-376.

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive science*, **12** (2), pp. 257-285.

Keywords: Heuristics; Prior knowledge; Problem solving; Strategies.

Author: Colin Foster, Mathematics Education Centre, Schofield Building, Loughborough University, Loughborough LE11 3TU.
e-mail: **c@foster77.co.uk**
website: **www.foster77.co.uk**