PROGRAMMING WITH DYSLEXIA

Tina Fountain and **Simon Carter** explore how dyslexia can be a strength when learning to program and give their tips for using that strength in teaching and learning

LEARNING TO PROGRAM AS A DYSLEXIC LEARNER

As a student in the classroom, my handwriting was illegible and my work was always incomplete. Note-taking from the board was impossible and I just could not keep up. I could understand the work and participate verbally, but I could never get my ideas down on paper. I still struggle with this today. I would work really hard on my written assignments, only to have them returned covered in red ink with comments on how I must try harder. When I was growing up, dyslexia was unheard of. By the time I was 13, I had simply stopped trying and left school at 16.

It wasn't until I entered the workforce that I began to thrive. Learning new skills came easily to me, my confidence grew, and I started a journey that led me back to education. Only then did I discover the term 'dyslexia', and suddenly every struggle I had experienced made sense.

I began my programming journey with HTML and CSS, and I was able to learn these with relative ease to the level I needed to teach them. Despite my initial confidence, learning to code beyond HTML and CSS was challenging. I could understand the basic principles and read other people's code, but I struggled with coding myself. Attending a Java course was one of the worst experiences of my life. The tutor expected us to copy her live coding as she went; it was like copying off the board at school, and I could not keep up. When I asked for help, she simply suggested that programming was not for me.

Luckily, I needed to learn to code for my job, so I dusted down my pride and searched for a way to teach myself. MIT's App Inventor (**appinventor.mit.edu**) was particularly helpful in building my skills and confidence, as it used a block-based development environment. Here, I was able to gain confidence and truly understand programming concepts. Learning text-based languages was then far more achievable. I am now confident in programming in multiple languages, and I finally conquered the basics of Java.

I found that writing out the syntax on index cards, along with simple applications of the concept, helped me immensely when learning Python. I needed time away from the screen to digest the requirements and to refer back to examples when tackling new problems. Planning on paper also helped me overcome the fear of the blank screen, and I now use this practice in the classroom with my students.

My dyslexia journey has been challenging, but it has also taught me the importance of understanding and accepting our unique learning styles. With perseverance and the right tools, anyone can learn to code.

TINA FOUNTAIN

Tina has over 20 years of teaching experience and currently teaches computing in a British school in Madrid. She is also part of the steering committee for the South East Asia Computer Science Teachers' Association (**@TinaFountain7**).

TOP TIPS TO SUPPORT DYSLEXIC LEARNERS

- Dyslexia can affect reading comprehension, which can make it difficult to understand complex instructions. Break down a task for students using bullet points rather than presenting paragraphed challenges.
- Dyslexia can also impact writing/typing skills, making writing and debugging code challenging. Provide a simple flowchart of debugging steps and lists of common error messages.
- Give students samples of code as a reference that they can use and adapt. Index cards are great for this purpose.
- Allow students time to annotate code and colour-coordinate variables or subroutine calls.
- Use videos for homework tasks. Videos will allow students to digest content in their own time and make notes without the pressure of keeping pace with the rest of the class.

ASSISTING DYSLEXIC PROGRAMMERS

Dyslexia is a condition that affects the incoming and outgoing signals by which our brain interacts with the world. It most commonly manifests itself as difficulties with reading, writing, processing auditory information, and performing arithmetic calculations.

We might expect that those with difficulties in reading and writing would have similar challenges in reading and writing computer programs. However, different and compensatory abilities make many dyslexic people exceptional programmers. Dyslexic programmers are helped by the fact that reading code is not like reading other forms of English. With only 33 keywords and 67 built-in functions in Python, for example, there are far fewer words to learn. Additionally, there is 100 percent consistency in the way keywords are used and spelled.

Dyslexic students also think visually, although not sequentially. They can perceive the various elements of a problem simultaneously, and so structure, hierarchy, and dependency can come more easily than to those who are not dyslexic. And therein lies probably the most powerful learning point I have come across in reading around this topic — that we should not lower our expectations of dyslexic programmers; we should merely expect different strengths. When they leave our classrooms, dyslexic programmers can become incredibly valuable systems analysts, members of software development teams, and software engineers. You can read more about why people with dyslexia make good programmers at **helloworld.cc/stein2018**.

> “ DIFFERENT AND COMPENSATORY ABILITIES MAKE MANY DYSLEXIC PEOPLE EXCEPTIONAL COMPUTER PROGRAMMERS

The choice and set-up of the programming environment, the IDE, is a key factor to consider. In assessing a programming assignment for a particular dyslexic student, for example, I was struck by two things. On the one hand, his coded solutions worked as well as anyone else's. Keywords and instructions were spelled accurately throughout. He told me that his IDE's IntelliSense feature (in Visual Studio), which gave him instant feedback about whether a word was spelled correctly and colour-coded keywords, was really important in his mastery of Python. This, combined with the frequency of use of these keywords, enabled him to achieve a high level of accuracy.

On the other hand, other aspects of his code were harder to follow because he was using one- or two-letter identifiers throughout. We teach students that identifiers should be descriptive and meaningful. He confirmed that he prefers identifiers to be as short as possible to minimise the likelihood of spelling errors and inconsistencies. It made complete sense to him and, on reflection, to me. We needed to work on developing a consistent and recognisable system, maybe using comment lines early on to list identifiers and their meanings in a way that would not cause errors later on. His code could still be self-documenting, but in a different way.

In her blog article '*Dyslexia and Coding*', Joanna Leng draws attention to the need to spend time selecting and customising a coding environment (**helloworld.cc/leng2017**). This will include setting up fonts (specialist fonts such as OpenDyslexic and Lexie Readable, or more standard sans serif fonts such as Verdana, Tahoma, Arial, and Calibri), text size (12–14pt), spacing between letters and lines, and background colours/themes such as dark text on a light non-white background.

Enabling tab completion in your IDE can add your identifiers to drop-down autosuggestions. Sometimes, new modules will need to be imported. There are VS Code extensions for spell-checking and colour-coding pairs of brackets (**helloworld.cc/Vsdyslexia**). It is important to remember that there is no one-size-fits-all solution. But if teachers have chosen an IDE that allows customisation, a little time spent personalising it for dyslexic students can make a world of difference to their experience. (HW)

SIMON CARTER

Simon is head of computing at Hardenhuish School in Wiltshire, UK. He has been teaching computing for over 25 years and is a founder and steering committee member of the South East Asia Computer Science Teachers' Association.