A Biographical Essay on Edsger Wybe Dijkstra

Edsger Wybe Dijkstra was born in the Netherlands on May 11th, 1930 and is one of the most influential people to have influenced the computer related field. It is highly likely that if you are studying computer science or learning how to code, at some point you will have heard the name "Dijkstra". From Dijkstra's early life it could be assumed that he was influenced by his father who was a chemist and his mother who was a mathematician. In the early 50's when even computer transistors were only being introduced Dijkstra by chance met director of the Computation Department at the Mathematical Center in Amsterdam. Very fortunately he was offered a job and become one of the computer programmers in the Netherlands. At first he envied the people who worked on the hardware/physical side of computation, wishing that he knew as much as they did about vacuum tubes and amplifiers compared to him. In his early computer related career, Dijkstra was still worried about whether he should move to pure physics instead of programming as it was such a new field. After a conversation with his mentor Adriaan van Wijngaarden, he made the decision to finish his study of physics and then stay in the emerging field as a computer programmer.

During his time at the Mathematisch Centrum of Amsterdam, he and two other colleagues had been hired to build a computer.
Before it was even actually being built, they were to decide on the interaction between hardware and software in the form
of a manual. After all was decided, they were to stick to their design or neither side could interface with the computer.
Dijkstra worked on software for this machine that had not yet existed. He learned valuable lessons on software engineering
and how clear documentation and careful design were essential in building his programs. During this time Dijkstra formulated
and solved the shortest path problem as one of the first demonstrations of the computer, this would go on to be one of his
more famous endeavors, having the solution named after him.

Another thing Dijkstra is well known for is his views on the "GO-TO" statement, after programming for several years he came to the conclusion that this statement led to poorly structured programs. In 1968 he wrote a private letter entitled "A Case against the GO TO Statement". After this letter was published publicly, a large debate within the programming community as to the statements viability. Since that time the GO-TO statement has fallen out of favour with many programmers and is barely used nowadays, or even completely disregarded in many modern programming languages. Dijkstra was a strong advocate of structured programming, that is, high level programs written in linearly structured forms. This way of programming brought many advantages as the code was a lot more standardized and could be easier read by others.

In 1965 Dijkstra introduced the 'mutual exclusion problem' for n processes and discussed a possible solution for it.
This is quite remarkable as at the time all computers basically ran one core at a time. Later on, in 1968 he discussed the notion of mutual exclusion or mutex. This was a way of locking each processor to work on a program without interference but still be able to combine their results after resulting in a faster computation than with just one processor. During this time he identified the deadlock problem where one CPU would wait for another, and the other was also waiting for the other. This would result in a deadlock where neither could do anything. To solve or get around this issue he proposed "Banker's algorithm".
He is credited as being one of the most influential and earliest pioneers of research on the idea of distributed computing, which is now used regularly and even phones have multi-core CPUs.

This and one of the other last categories of computer science Dijkstra would study was formal verification. His focus was mostly on the mathematics of programming and non-determinism. He used a language he called “Guarded Command Language” or GCL, which could be used for proofs and its development method is still seen today.

It is very difficult to measure the absolute impact that a person or a person's idea has had on any field, as we have no idea how many people might have been directly inspired or been able to solve a problem they have had easier because of an idea. It is as difficult if not more so to guess the impact the lack of an idea or concept would have been and if that idea would have been developed later, or not at all. Despite this after reading up so much on Edsger Wybe Dijkstra, I believe he has made one of the greatest impacts on this field of computer science. Only a few dozen people could ever be compared to what this man achieved in his lifetime.

Lastly, I personally am very happy to have read about his impact on computer science, changing in my mind from a person with a very difficult name to spell and pronounce that made some path finding algorithm, to a truly influential figure to be looked up to and inspired by. (Plus he has the same birthday as me).

Sources:
https://www.cs.utexas.edu/users/EWD/MemRes(USLtr).pdf
https://homepages.cwi.nl/~apt/ps/dijkstra.pdf
https://www.cs.utexas.edu/users/EWD/transcriptions/EWD05xx/EWD563.html