Why do we refer to Programming Principles, Logic and Verification (PPLV) as the heart of computer science?

The answer is quite simple. If we imagine computer science to be a tree, subjects such as artificial intelligence or graphics and interaction would be the branches, whereas PPLV would be the trunk.

Why is that?

PPLV is a way for programmers to analyse and create programs (Programming Principles), create mathematical and algorithmic solutions and concepts to create programs or use in other fields (Logic), and finally to verify the validity, efficiency, safety, etc .. of said program (Verification).

Hence we can conclude that essentially almost – if not – all computer science branches use PPLV in one way or another.

But how has the PPLV department at UCL set its impact on the world?

Out of the many projects, Facebook Infer is one of the more known successes of the PPLV department. What exactly is Facebook Infer? It is a collaboration between the PPLV department and Facebook where a program was created to identify bugs within other source codes. This may seem small, you may be thinking, identifying bugs can be simple with a little search through one’s code. As codes for larger and larger projects grow, so does the potential for bugs, and so does the difficulty to identify them manually. Hence the Facebook Infer is a very important and useful tool for high-level programmers.

<http://fbinfer.com/> - In case you’re interested to learn more!

Another impact of the PPLV department was the development of the “Productive Security”. This project is a framework where it takes as an input a company’s / organization’s workload and risk perception and tells the programmers what sort of security measures is most efficient taking into consideration the input it was given. This can be extremely helpful for security purposes as not all organizations hold the same type of information / data structure and thus will require different measures of security. As obvious as it sounds, this project was aimed to higher the security compliance of organizations around the globe.

There are numerous other projects that aid programmers or other users with the verification of their systems and / or programs. For example the Static Driver Verifier which is used in Microsoft to discover defects and design issues in a driver. (This comes under the verification part of PPLV). Another example is “Bunched Logic” developed by two academic staff at the PPLV department, which helps with compositional analysis of computers and other systems through reasoning about resource composition.

<https://en.wikipedia.org/wiki/Bunched_logic> - In case you’re interested to learn more! (Although this is very technical and probably best not to search up)

PPLV stands to be one of the most impactful departments in computer science as it is the core of the subject, taking into consideration that its three fundamentals, Programming Principles, Logic, and Verification are almost used interchangeably in every system and / or program used in today’s world. What happens in PPLV is the pillars on which these programs and systems are built on.