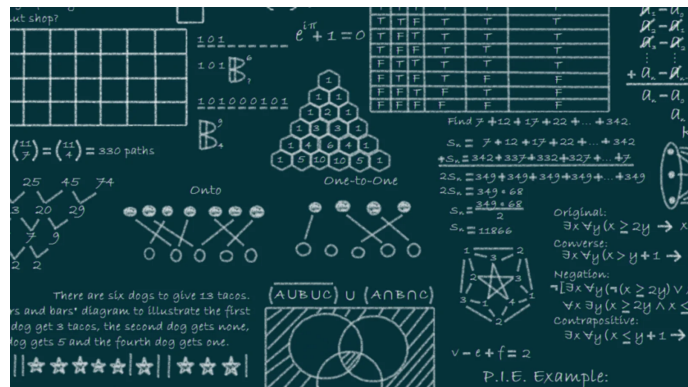




Algorithms, Data Structures, Complexity Theory — Bring Your Own Idea

Are you excited about a problem in algorithms, data structures, or just the theory of computation in general? Behind most modern-day industrial applications stand sophisticated models for efficient data processing, combinatorial optimization and reasoning.

At the core of theoretical CS stand two governing questions: can it be done? and, if so, how efficiently? in terms of runtime, space complexity or perhaps other metrics, like the number of oracle calls or communication complexity (how many bits need to be exchanged to achieve a goal).



Here, at DISCO, we are equally interested in theory and applications, so we are willing to supervise projects with any of the following goals (but not limited to):

1. Design a new algorithm/data structure with novel guarantees. Or, maybe you can find an approach that is more elegant or significantly more efficient in practice than what is currently known? Sequential, concurrent and distributed settings all qualify.
2. Tighten the analysis of an existing approach. E.g. did you know that the best minimum spanning tree algorithm is known, but its worst case runtime is yet to be figured out?
3. Proving new algorithmic lower bounds. In other words, show impossibility results. E.g. there is an $O(n)$ distributed algorithm for diameter computation, but so far we only know that one can not do better than $O(n/\log n)$, leaving a gap worth studying.
4. Consider a non-standard model of computation and prove something cool about it.

Requirements. Ability to work independently and interest in conducting new research. Solid algorithmic and mathematical background (emphasis on writing proofs). Depending on the extent of the project, coding skills are a plus. We will have weekly meetings to discuss open questions and determine the next steps.

Interested? Please contact us for more details!

Contact

- Andrei Constantinescu: aconstantine@ethz.ch, ETZ G93