Constraint Programming
Quiz #09 (advanced search and optimization)

What is the major difference between iterative broadening and other incomplete tree search techniques such as depth-bounded search?

Can depth-bounded search be simulated using credit search?

What are the properties of value ordering heuristics that are exploited by the LDS algorithm?

What is a discrepancy?

What are the major motivations behind ILDS in comparison to LDS?

Does LDS revisit the branches in future iterations?

Does ILDS revisit the branches in future iterations?

How does the depth-bounded discrepancy search ensure that the algorithm does nor re-visit the same branch in future iterations?

Which property of constraint satisfaction algorithms do we need to solve CSOPs?

How is the heuristic cost estimate implemented when solving CSOPs?

What is the practical difference between classical branch-and-bound and dichotomic branch-and-bound algorithms?

What is the difference between finding an optimal solution and knowing that the solution is optimal?