

## Research Projects

### Inferring Grammars

PROF. SUBHAJIT ROY

#### Compilers, First Order Logic

*Summer 2016*

The project was aimed at inferring grammars from a given set of strings known to be correct or incorrect in that grammar. This had been implemented for LL(1) grammars and my work was aimed at further optimizing it and working to extend the same for error correction or similar tasks for LR(k) grammars. In essence this project was aimed at building AI for Inferring Grammars and hence a step along the way to enable us to infer grammar of mainstream languages and work on optimizing compilers for the same.

### MaxSAT optimization

PROF. SUBHAJIT ROY

#### NP Hard problems, Parallel Programming

*Summer 2016*

The project was aimed at solving large instances of MaxSAT by converting them to instances of Set Cover using a parallelized algorithm optimized with due considerations, written in CUDA C. That would enable us to convert the NP Hard problem into an instance of a NP complete solving of which is done in well bound times by efficient algorithms commonly known. Better algorithms have a bound on normal processors but we can work better if we use the power of GPUs to solve problems.

---

## Course Projects

### Online Judge and Game

PROF. SUNIL E. SIMON

#### Functional Programming, Game Theory

*December 2015 - April 2016*

This project was aimed at building an Online Judge for judging programs written in C/ Haskell in a sandbox which also made sure that it not only checked for safe code but had inbuilt fail-safe in case there was an instance of faulty or bad code. It handled all errors and returned them to the dynamic web portal which allowed the users to submit solutions to problems and admins to add problems and view the solutions submitted by individuals. The project also involved using Lambda Calculus to solve an instance of TicTacToe more efficiently and hence optimize known algorithms further.

### Grammar Checking Tool

PROF. SUNIL E. SIMON

#### Functional Programming, Formal Grammars

*December 2015 - April 2016*

This project was aimed at building a system written in Haskell which aimed to check if a given grammar given as input in form of its productions, terminals and non terminals; was first Context Free or not and then LL(1) or not. This was more a personal project undertaken since there was no easily available pieces of code that would do that. Further this was done dynamically but not by abstraction which made sure that the complexity of giving this result would be polynomial time in the size of the Grammar and not exponential which it would have been in case of abstracting out grammars as Logic or if naive algorithms were used in the regime of Imperative Programming.

### Talk on Error Correcting Codes

PROF. NITIN SAXENA

#### Discrete Mathematics

*November 2016*

The talk was aimed at familiarising the audience with Block codes, Linear Codes and theory which gave approximate upper bounds on the error. This also gives us bounds on how much data can be transferred and how we can decode data with minimal losses. Further, we can notice how this is applied very efficiently in nature in form of Protein Translation or Transcription from DNA.

---

## ExtraCurricular Projects

### Autonomous Underwater Vehicle

PROF. K. VENKATESH

#### Robotics

*October 2015 - June 2016*

This project was aimed at developing an underwater Vehicle which is not only aware of it's surroundings but autonomous in a pre-programmed way. The aim was to complete a set of tasks in random order detected according to environment. The work was aimed at making the Bot mechanically able to complete all such tasks, but moreover enhance and configure control algorithms for the motion of the Bot and use learning techniques to achieve Image Processing for the bot to be able to do these tasks. In addition to this high level control structure must be generated and programmed specifically using Robot Operating Systems to achieve this. My work was mainly limited in the control algorithms and making the Bot mechanically able regime.