Anthonys Thesis Roadmap

Name: Anthony Kostal-Vazquez
Student ID: 30048301

Goal 0 — Understand basic binary quadratic form background

(a)~~Read Antons thesis~~ Thesis Link

(b)Read Johannes Buchmann text Textbook Link

(c)~~Show a reasonable certainty that the class number generating formulas will be an improvement~~

Goal 1 — Extend Polymult 1.4 to handle mod 24, mod 120, mod 720 and beyond

(a)Understand how polymult 1.4 works

- ~~Learn to code in C~~
- Read about polymult 1.3 in Antons thesis Thesis Link
- Read polymult 1.3 documentation Decumentation Link
- Read polymult 1.4 documentation Decumentation Link
- ~~Get running on local machine~~
- ~~Get running on ARC machine~~

(b)Understand the class number formulas

- Take Algebra 1 course equivalent (Link to syllabus)
- Read about original mod 8 formulas in Antons thesis Thesis Link
- Read Bringmann and Kane paper (Need link to this)
- Understand how the mod 24/120 formulas where created Link to paper
- Create the mod 720 formulas
- Test mod 120 and mod 720 formulas with robust and probably slow python polynomial multiplication program for correctness.

(c)Implement formulas into polymult 1.4

- Make new formulas compatible with bounds that are powers of two
- Give polymult the ability to initialize then add polynomials since this is required for the mod 120 formulas
- Figure out why the modulo 120 formulas will not work with polymult 1.4
- Turn excel initialization routines into a scripting language (Link to excel sheets)
- Figure out why polymult 1.4 has such low cpu utilization on ARC
- Test at scale

(d)Polish and publish library publicly

- Generalize so that polymult 1.4 can multiply not just theta and nabla series
- Create documentation (basically extension of polymult 1.3 documentation)

Goal 2 — Implement Sutherlands algorithm in Clgrp 1-3

(a) Understand how Clgrp 1.3 works

- Take Algebra 1 course equivalent (Link to syllabus)
- Read about clgrp algorithms in Antons thesis Thesis Link
- Read Clgrp Documentation Documentation Link
- ~~Get running on local machine~~
- ~~Get running on ARC machine~~

(b) Ensure compatibility between polymult 1.4 library and Clgrp 1.3

- Ensure that clgrp is able to take in the outputs of polymult
- ~~Fix the chinese remainder theorem subroutine so that m or n dont have to be powers of 2~~

(c) Ensure compatability with mod 120 and mod 720 formulas

- Make sure that n = 120 and n = 720 doesn't break anything
- Fix the tabulation of 23mod120
- Fix the memory leak causing clgrp 1.3
- Extend the verification to work with the new modulo breakdown

(d) Learn what Sutherlands algorithm is

(e) Implement Sutherlands algorithm into clgrp 1.3

(f) Test Large scale on ARC (Specifically the h3mod8 tabulations seem to be tricky)

Goal 3 — Run tabulations beyond 2^{40}

(a) Small scale testing

(b) Recreation of 2^{40} result on ARC

- ~~Take the ARC class~~
- ~~learn how to use slurm scripts~~

(c) Run beyond 2^{40} on ARC

Goal 4 — Analyze Results

(a) Find out how this was done for Antons tabulation

Goal 5 — Write Thesis

(a) Learn to use latex

Goal 6 — Defend Thesis