Проекты по практикуму по компьютерной алгебре

1. Реализовать алгоритм F4. Алгоритм F4 был предложен Ж.-Ш. Фожером в 1999 г. Этот алгоритм вычисляет базис Гребнера идеала в кольце многочленов с помощью серии стандартных линейноалгебраических процедур: приведений матриц к ступечатому виду. Он является одним из самых быстрых на сегодняшний день.

Пусть есть некоторое конечное множество многочленов F. По этому множеству строится матрица, строки которой соответствуют многочленам из F, а столбцы — мономам. В матрице записаны коэффициенты многочленов при соответствующих мономах. Столбцы матрицы упорядочены согласно выбранному мономиальному упорядочению (старший моном соответствует первому столбцу). Приведение такой матрицы к ступенчатому виду позволяет узнать базис линейной оболочки многочленов из F в пространстве многочленов.

Пусть в классическом алгоритме Бухбергера требуется провести шаг редукции многочлена f относительно g, и при этом g должен быть домножен на моном M. В алгоритме F4 в матрицу будут специально помещены f и Mg. Утверждается, что можно заранее подготовить множество всех потенциальных домноженных редукторов, которые могут потребоваться, и поместить их заранее в матрицу. Кроме того, вместо S-полиномов можно поместить в матрицу их левые и правые части (при редукции одной строки по другой автоматически получится S-полином). Наконец, третьим отличием от алгоритма Бухбергера является то, что в алгоритме F4 разрешается поместить в одну матрицу части сразу нескольких S-полиномов, выбранных согласно какой-либо стратегии.

Требуется реализовать алгоритм F4.

Подробнее:

1. J.-C. Faugère *A New Efficient Algorithm for Computing Gröbner Bases (F_4).* Journal of Pure and Applied Algebra, 139 (1999), 61–88. http://www-calfor.lip6.fr/~jcf/Papers/F99a.pdf
2. B. H. Roune. *The F_4 algorithm.* http://www.broune.com/papers/f4.pdf

2. Реализовать матричный алгоритм F5. Алгоритм F5 вычисления базиса Гребнера был предложен Ж.-Ш. Фожером в 2002 году. Предлагается реализовать его матричную версию (в духе алгоритма F4), работающую для однородных многочленов. Основная процедура этого алгоритма вычисляет D-базис Гребнера, то есть, подмножество базиса Гребнера, относительно которого редуцируются к нулю все многочлены из идеала степени не выше, чем D.

В алгоритме F5 каждому полученному многочлену сопоставляется *сигнатура* (пара из монома и номера образующей), кодирующая информацию о происхождении этого многочлена. Основная идея — не включать по возможности в матрицы те строки, которые будут линейно зависимы с остальными (то есть, будут редуцироваться к нулю.)

Подробнее:

1. J.-C. Faugère. *Habilitation à diriger des recherches.* http://ebookbrowse.com/new-hdr-faugere-pdf-d292014224
2. J.-C. Faugère, S. Rahmany. *Solving Systems of Polynomial Equations with Symmetries Using SAGBI-Gröbner Bases.* §3.3. http://www-polsys.lip6.fr/~jcf/Papers/ISSAC09a.pdf
3. J.-C. Faugère, M. El Din, P.-J. Spaenlehauer. *Gröbner Bases of Bihomogeneous Ideals Generated by Polynomials of Bidegree* $(1,1)$*: Algorithms and Complexity.* Page 5. http://www-polsys.lip6.fr/~jcf/Papers/JSC_FSS10.pdf
4. J.-C. Faugère. *A new efficient algorithm for computing Gröbner bases without reduction to zero (F5).* http://www-salsa.lip6.fr/~jcf/Papers/F02a.pdf

3. Реализовать алгоритм Бухбергера для модулей над кольцами многочленов. Пусть $P = F[x_1, \ldots, x_n]$ — кольцо многочленов над полем. Рассмотрим свободный модуль $\mathcal{P} = P^m$ над этим кольцом. Пусть $e_1, \ldots, e_n$ — его базис. Всякий элемент из $\mathcal{P}$ имеет вид $\sum_{i=1}^{m} f_i e_i$. Можно естественным образом расширить понятие допустимого порядка на множество *термов*, то есть, выражений вида $x_1^{a_1} \ldots x_n^{a_n} e_i$. Тогда для подмодуля $N \subset \mathcal{P}$ можно ввести понятие базиса Гребнера. Реализуйте алгоритм Бухбергера для вычисления этого базиса Гребнера.

Подробнее:

1. M. Kreuzer, L. Robbiano. *Computational Commutative Algebra (Part 1).* §2.4. http://lib.mexmat.ru/books/1106

4. Алгоритм Бухбергера для произвольного упорядочения. Известно, что всякое мономиальное упорядочение на n переменных можно задать матрицей M размера $m \times n$ следующим образом:

$$x_1^{a_1} \dots x_n^{a_n} \prec x_1^{b_1} \dots x_n^{b_n} \iff M \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} <_{lex} M \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}.$$

При этом матрица M должна удовлетворять следующим свойствам: ее столбцы должны быть лексикографически неотрицательны, а множество рациональных решений соответствующей однородной системы линейных уравнений должно быть нулевым. Например, лексикографическое упорядочение можно задать единичной матрицей.

Требуется реализовать алгоритм Бухбергера, который вычислял бы редуцированный базис Гребнера для произвольного упорядочения, заданного указанной матрицей.

5. Алгоритм FGLM. Многочисленные практические вычисления показывают, что базис Гребнера при порядке degrevlex вычисляется обычно быстрее, чем базис Гребнера при порядке lex. В случае, когда идеал нульмерен (то есть, в алгебраическом замыкании поля существует лишь конечное множество решений соответствующей системы уравнений), можно эффективно преобразовывать базисы Гребнера от одного упорядочения к другому с помощью линейной алгебры. Алгоритм такого преобразования предложили в 1994 году Faugère, Gianni, Lazard и Mora.

Алгоритм работает в конечномерной факторалгебре $A = k[x_1, \dots, x_n]/I$ и пополняет список мономов B и многочленов G. Основной цикл алгоритма берет очередной в лексикографическом порядке моном M, рассматривает его образ $\bar{M}$ в A и пытается проверить его на линейную зависимость с лексикографическими остатками мономов из B относительно G. Если M линейно зависим, то уравнение его линейной зависимости дает новый элемент лексикографического базиса. Оно добавляется к множеству G. В противном случае M добавляется ко множеству B. Алгоритм останавливается, когда получится уравнение линейной зависимости со старшим мономом вида x_1^m.

Подробнее:

1. D. Cox, J. Little, D. O'Shea. *Using Algebraic Geometry,* Chapter 2, §3. http://lib.mexmat.ru/books/54134
2. J.-C. Faugère, P. Gianni, D. Lazard, T. Mora. *Efficient Computation of Zero-Dimensional Gröbner Bases by Change of Ordering.* Journal of Symbolic Computation 16 (4): 329–344 (1994). http://www-polsys.lip6.fr/~jcf/Papers/FGLM.pdf

6. Автоматическое доказательство геометрических теорем. Пусть имеется некоторое элементарное геометрическое утверждение в Евклидовой геометрии на плоскости или в пространстве (например: заданные отрезки перпендикулярны; длины отрезков равны; скалярные произведения векторов равны и т.д.). Введем удобным образом декартову систему координат. Запишем это утверждение в виде полиномиального уравнения от координат (если это возможно).

Пусть теперь дана некоторая геометрическая теорема, исходные данные и утверждение которой переводятся на полиномиальный язык. Пусть F — уравнения, задающие ее условия, а f — уравнение, задающее то утверждение, которое надо доказать. Пусть также через h обозначено уравнение, задающее вырожденные случаи (например, три точки оказались на одной прямой). Тогда если $fh \in \sqrt{\langle F \rangle}$, то теорема верна в невырожденном случае.

Требуется написать библиотеку функций, превращающих элементарные геометрические утверждения в уравнения, а также проверяющих истинность теорем.

Подробнее:

1. Д. Кокс, Дж. Литтл, Д. О'Ши, *Идеалы, многообразия и алгоритмы,* М., Мир, 2000. Глава 6, §4. http://lib.mexmat.ru/books/1085, http://lib.mexmat.ru/books/36731
2. S.-C. Chou, *Mechanical Geometry Theorem Proving,* D. Reidel Publishing Company, Dordrecht, 1988. http://lib.mexmat.ru/books/1021

7. Универсальный базис Гребнера. Хотя множество допустимых упорядочений на мономах от нескольких переменных имеет мощность континуума, существует лишь конечное число редуцированных базисов Гребнера данного идеала. Если объединить эти базисы, то получится *универсальный базис Гребнера*, который является (нередуцированным) базисом Гребнера при любом упорядочении.

Алгоритм построения универсального базиса легко получить из алгоритма Бухбергера. Действительно, на каждом шаге у каждого многочлена имеется лишь конечное число потенциальных старших мономов.

Требуется написать алгоритм проверки того, что данное множество является универсальным базисом Гребнера данного идеала.

Подробнее:

1. T. Becker, V. Weispfenning. *Gröbner Bases,* p. 514. http://lib.mexmat.ru/books/1110

8. Маршрут Гребнера (Groebner Walk). Алгоритм Groebner Walk позволяет аналогично алгоритму FGLM преобразовывать базисы Гребнера от одного упорядочения к другому без полного пересчета, но он основан на других идеях и его можно применять к произвольным идеалам (а не только к конечномерным). Основная идея заключается в том, что множество всех допустимых упорядочений разбивается для данного идеала на конечное число подмножеств (Groebner Fan), в каждом из которых редуцированный базис Гребнера идеала один и тот же. Алгоритм Groebner Walk пытается построить путь из одного подмножества к другому, пересчитывая базис Гребнера на каждой границе.

Требуется реализовать алгоритм Groebner Walk.

Подробнее:

1. D. Cox, J. Little, D. O'Shea. *Using Algebraic Geometry,* Chapter 8, §5. http://lib.mexmat.ru/books/54134
2. S. Collart, M. Kalkbrener, D. Mall. *Converting Bases With the Gröbner Walk.* J. Symbolic Computation (1997) 24, 465–469. https://www.risc.jku.at/research/theorema/Groebner-Bases-Bibliography/gbbib_files/publication_637.pdf

9. Алгоритм шифрования HFE. Пусть $\mathbb{K} = \mathbb{F}_{p^m}$ — основное поле. Его элементы станут «буквами» кодируемого сообщения. Пусть длина нашего сообщения равна n. Рассмотрим расширение $\mathbb{L} = \mathbb{F}_{p^{mn}}$ исходного поля $\mathbb{K}$. Тогда $\mathbb{L}$ является векторным пространством над $\mathbb{K}$ размерности n. Пусть $\varphi : \mathbb{K}^n \to \mathbb{L}$ — биекция векторных пространств. Зафиксируем также две невырожденные аффинные замены координат A и B: $\mathbb{K}^n \to \mathbb{K}^n$. В общем случае это умножение на обратимую $n \times n$ матрицу и перенос на некоторый вектор.

Пусть $f(z) \in \mathbb{L}[z]$ имеет вид

$$f(z) = \sum \alpha_{s,t} z^{p^{ms}+p^{mt}} + \sum \alpha_s z^{p^{ms}} + \alpha,$$

где $\alpha_{s,t}, \alpha_s, \alpha$ — некоторые параметры.

Получаем схему шифрования с открытым ключом. Секретный ключ — знание A, f, B по отдельности. Открытый ключ — композиция

$$A\varphi^{-1} f \varphi B : \mathbb{K}^n \to \mathbb{K}^n,$$

которая будет задаваться многочленами степени не выше второй по каждой координате. Если обозначить исходное сообщение строкой $(a_1, \ldots, a_n)$, то кодом будет строка

$$(f_1(a_1, \ldots, a_n), \ldots, f_n(a_1, \ldots, a_n)),$$

где $f_i(a_1, \ldots, a_n)$ — квадратичные многочлены от многих переменных.

Взлом системы заключается в решении системы n квадратичных уравнений от неизвестных $a_1, \ldots, a_n$ над полем $\mathbb{K}$.

Требуется написать программы, которые по заданным параметрам генерировали бы открытый и секретный ключи, зашифровывали и расшифровывали сообщения по указанным ключам, а также пытались бы взломать шифр, решив систему с помощью вычисления базиса Гребнера.

Подробнее:

1. J. Patarin. *Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): two new Families of Assymetric Algorithms.* http://www.minrank.org/hfe.pdf
2. N. Courtois. *On multivariate signature-only public key cryptosystems.* http://eprint.iacr.org/2001/029.pdf
3. *Hidden Field Equations.* http://en.wikipedia.org/wiki/Hidden_Field_Equations

10. Работа с дифференциальными многочленами. Пусть $F\{y\} := F[y_0, y_1, \ldots, y_n, \ldots]$ — кольцо многочленов над полем F от счетного числа переменных. Введем на нем оператор дифференцирования δ по следующему правилу:

— $\delta c = 0$ для $c \in F$;
— $\delta y_k = y_{k+1}$;
— $\delta(f + g) = \delta f + \delta g$;
— $\delta(fg) = \delta f\, g + f \delta g$.

Получилось *кольцо дифференциальных многочленов* от одной независимой переменной y с одним дифференцированием δ. Это определение можно обобщить на случай нескольких независимых переменных и нескольких дифференцирований.

Порядком переменной y_k называется число k. Порядок многочлена равен максимуму из порядков его переменных.

В случае нескольких дифференциальных переменных для определения их старшинства требуется ввести *ранжир* на множестве всех переменных, то есть, полный порядок, удовлетворяющий свойствам

1. $u < w \implies \delta u < \delta w$;
2. $1 \leqslant u$.

Необходимо написать конструкции в Sage для работы с дифференциальными многочленами (по аналогии с обычными многочленами). В частности, должны быть предусмотрены функции

— `diff` (применение дифференцирования к многочлену);
— `ord` (порядок многочлена);
— `leader` (старшая переменная и ее максимальная степень);
— `initial` (полиномиальный коэффициент перед лидером);
— `separant` (частная производная по старшей переменной, то есть, инициал любой производной этого многочлена).

Подробнее:

1. W. Sit. *The Ritt-Kolchin Theory for Differential Polynomials*. Google Books.
2. E. Kolchin. *Differential Algebra*. http://lib.mexmat.ru/books/1014.

11. Алгоритм дифференциальной псевдоредукции. Пусть $F\{y\}$ — кольцо дифференциальных многочленов от одной переменной с одним дифференцированием над полем нулевой характеристики (см. предыдущую задачу). Если зафиксирован ранжир на дифференциальных переменных, то у каждого ненулевого дифференциального многочлена f можно определить старшую переменную u_f и ее максимальную степень d_f. Пусть I_f и S_f — инициал и сепаранта дифференциального многочлена f. Определим процесс дифференциальной редукции. Если в f есть слагаемое, кратное $u_g^{d_g}$, то надо f домножить на I_g и вычесть g с подходящим множителем. Если же в f есть слагаемое, зависящее от производной θu_g, то надо f домножить на S_g и вычесть θg с подходящим множителем.

Напишите процедуры редукции дифференциального многочлена по заданному множеству, а также процедуры проверки того, что данное множество авторедуцировано.

Подробнее:

1. W. Sit. *The Ritt-Kolchin Theory for Differential Polynomials*. Google Books.
2. E. Kolchin. *Differential Algebra*. http://lib.mexmat.ru/books/1014.

12. Некоммутативные многочлены. Требуется создать инфраструктуру для работы с некоммутативными многочленами (элементами свободной ассоциативной алгебры) от нескольких переменных с коэффициентами из некоторого поля в Sage. С помощью написанных классов или функций требуется решить следующую задачу. Пусть $x_1, \ldots, x_n$ — некоммутативные переменные. Зададим правила дифференцирования: $\delta x_k = p_k(x_1, \ldots, x_n)$, где p_k — некоммутативные многочлены. Распространим это дифференцирование δ на все многочлены по линейности и правилу Лейбница. Требуется написать функцию дифференцирования некоммутативного многочлена в силу заданных правил.

13. Разложение на примарные компоненты. Пусть $I = \langle f_1, \ldots, f_m \rangle \lhd F[x_1, \ldots, x_n]$ — нульмерный идеал (то есть, соотвествующая система алгебраических уравнений имеет лишь конечное число решений V в $\overline{F}$). Требуется реализовать алгоритм, раскладывающий I в пересечение примарных идеалов.

Предположим, что все x_1-компоненты решений V различны. В этом случае радикал идеала I имеет вид

$$\sqrt{I} = \langle g_1(x_1), g_2(x_1) - x_2, \ldots, g_n(x_1) - x_n \rangle.$$

Построим с помощью лексикографического базиса Гребнера образующую p идеала $I \cap F[x_1]$. Пусть $p = p_1^{d_1} \ldots p_s^{d_s}$ — разложение на линейные множители над $\overline{F}$. Тогда примарные компоненты I это в точности идеалы $I + \langle p_k^{d_k} \rangle = \langle f_1, \ldots, f_m, p_k^{d_k} \rangle$.

Пусть теперь I — произвольный нульмерный идеал. Пусть z — новая переменная. Можно подобрать коэффициенты $c_2, \ldots, c_n$ так, чтобы для идеала $J = \sqrt{I} + \langle z - x_1 - c_2 x_2 - \ldots - c_n x_n \rangle$ все z-компоненты решений были бы различны. Тогда примарное разложение I получается из степеней ассоциированных простых компонет J, пересеченных с $F[x_1, \ldots, x_n]$.

Подробнее:

1. T. Becker, V. Weispfenning. *Gröbner Bases,* p. 366–388. http://lib.mexmat.ru/books/1110
2. G.-M. Greuel, G. Pfister. *A Singular Introduction to Commutative Algebra*, §4. http://lib.mexmat.ru/books/35216, PDF

14. Простое разложение нульмерного радикального идеала. Идея аналогична предыдущей: все примарные компоненты радикального идеала являются простыми. Требуется лишь убедиться, что данный нульмерный идеал действительно является радикальным.

15. Проверка примарности нульмерного идеала. Пусть $I \lhd F[x_1, \ldots, x_n]$ — нульмерный идеал. Если он находится в общем положении относительно переменной x_n (то есть, x_n-координаты решений соответствующей системы не повторяются), то его примарность эквивалентна тому, что

1. $I \cap F[x_n] = \langle g_n(x_n)^{d_n} \rangle$, где многочлен g_n неприводим,
2. для всех $j < n$ в идеале I есть многочлен вида $(x_j + g_j(x_n))^{d_j}$.

Реализуйте алгоритм проверки примарности нульмерного идеала.

Подробнее:

1. G.-M. Greuel, G. Pfister. *A Singular Introduction to Commutative Algebra*, §4.2 http://lib.mexmat.ru/books/35216, PDF

16. Вычисление характеристических многочленов. Пусть R — некоторое кольцо, A — матрица $n \times n$ над этим кольцом. Требуется эффективно вычислить характеристический многочлен матрицы A. Прямое вычисление определителя $\det(A - \lambda E)$ не является эффективным (кольцо R может не быть полем).

Предлагается вычислить характеристический многочлена методом Леверье. Сначала вычисляются все степени матрицы: $A^2, A^3, \ldots, A^n$. Для каждой степени вычисляется след: $s_k = \operatorname{tr} A^k$. Далее коэффициенты характеристического многочлена вычисляются с помощью формул Ньютона, связывающих степенные суммы и элементарные симметрические многочлены (см. задачу 3 из раздела «Многочлены» третьей домашней работы).

Так как для вычисления следа достаточно знать только диагональные элементы матрицы, то можно вычислять не все степени матриц, а лишь некоторые. А именно, сначала прямо вычисляются $A^2, \ldots, A^m$ при $m \leqslant n$, а затем $A^{2m}, A^{3m}, \ldots, A^{[n/m]m}$. Этих матриц достаточно, чтобы найти диагональные элементы остальных степеней матрицы A. Обычно выбирают $m \approx \sqrt{n-1}$, что дает лучшую оценку сложности. Такое усовершенствование называется алгоритмом Леверье-Винограда.

Реализуйте методы Леверье и Леверье-Винограда.

Подробнее:

1. Massoud Malek. *Characteristic Polynomial.* http://www.mcs.csueastbay.edu/~malek/Class/Characteristic.pdf

17. Базисы Гребнера и раскраски графов. Пусть дан конечный граф с множеством вершин V и множеством ребер E. Требуется каждую вершину раскрасить в один из n цветов, чтобы соседние вершины были раскрашены в разные цвета. Эту задачу можно перевести на язык систем алгебраических уравнений и решить с помощью базисов Гребнера.

Пусть каждой вершине k соответствует своя переменная x_k. Рассмотрим систему следующих уравнений над $\mathbb{C}$:

- $x_k^n - 1$ для каждой вершины $k \in V$;
- $x_i^{n-1} + x_i^{n-2}x_j + \ldots + x_i x_j^{n-2} + x_j^{n-1}$ для каждого ребра $(i, j) \in E$.

Компоненты решений этой системы состоят из комплексных корней степени n из единицы (кодирующих цвет). Каждое решение взаимно однозначно соответствует раскраске графа в n цветов (докажите!). Реализуйте алгоритм, который по заданному графу и числу n находит его раскраску в n цветов (или сообщает, что раскраски не существует), решая эту систему с помощью базиса Гребнера.

1. C. Hillar, T. Windfeldt. *An Algebraic Characterization of Uniquely Vertex Colorable Graphs.* Citeseer.
2. C. Hillar. *Gröbner Bases and Applications.* Presentation.

18. Многочлен с разреженным квадратом. Постройте многочлен f с рациональными коэффициентами, такой, что в развернутой записи f^2 слагаемых меньше, чем в самом многочлене f. Найдите такой многочлен минимально возможной степени. Как это сделать с помощью базисов Гребнера? Например, так. Будем итеративно увеличивать степень d текущего многочлена, записанного с неопределенными коэффициентами. Можно считать, что старший коэффициент многочлена — единица. Тогда можно написать много отдельных систем уравнений, выражающих то, что в f^2 определенные коэффициенты зануляются, а затем проверить, есть ли среди них совместная система.

Реализуйте записать алгоритм, который для заданного $\varepsilon > 0$ находит многочлен f минимальной степени, такой, что $|\mathrm{Supp}(f^2)| < \varepsilon|\mathrm{Supp}(f)|$, где $|\mathrm{Supp}(f)|$ – число слагаемых в развернутой записи многочлена.

1. M. Kreuzer, L. Robbiano. *Computational Commutative Algebra (Part 1).* Tutorial 42. http://lib.mexmat.ru/books/1106

19. Решения судоку. Решение логической игры судоку можно найти аналитически, сведя эту задачу к задача о раскраске графа девятью красками и переписав это условие в виде системы полиномиальных уравнений. Ваша задача — написать программу, которая по заданной начальной комбинации судоку выписывала бы такую систему уравнений и пыталась бы найти решение.

1. H. Gago-Vargas, I. Harillo-Hermoso, J. Martín-Morales, J. M. Ucha-Enríquez. *Sudokus and Gröbner bases: not only a Divertimento.* http://www.ricam.oeaw.ac.at/Groebner-Bases-Bibliography/gbbib_files/publication_1180.pdf
2. A. Steenpass. *SINGULAR Tutorial.* Exercise Sheet 3. http://math.ipm.ac.ir/conferences/2011/cca2011/Exercises-Steenpass.pdf

20. Число Фробениуса. Пусть $p = (p_1, \ldots, p_n)$ — набор попарно взаимно простых натуральных чисел. Число $n \in \mathbb{N}$ будем называть p-представимым, если n можно представить в виде линейной комбинации $\lambda_1 p_1 + \ldots + \lambda_n p_n$ для некоторых неотрицательных целых λ_i. Числом Фробениуса f_p^* называют максимальное целое число, которое не является p-представимым. Например, для $p = (6, 10, 15)$ число Фробениуса равно 29. Ваша задача — вычислить число Фробениуса для заданного p. Оказывается, это можно сделать сравнительно эффективно с помощью базисов Гребнера.

1. B. Roune. *Solving Thousand Digit Frobenius Problems Using Gröbner Bases.* Journal of Symbolic Computation, vol. 43, issue 1, 2008. http://www.broune.com/papers/compFrob.pdf
2. N. Lauritzen. *Gröbner Bases and the Frobenius problem.* http://math.au.dk/fileadmin/Files/matlaererdag/2005/niels.pdf

21. Целочисленное программирование. Рассмотрим диофантову систему линейных уравнений

$$\begin{cases} a_{11}\sigma_1 + a_{12}\sigma_2 + \ldots + a_{1n}\sigma_n = b_1, \\ a_{21}\sigma_1 + a_{22}\sigma_2 + \ldots + a_{2n}\sigma_n = b_2, \\ \ldots \\ a_{m1}\sigma_1 + a_{m2}\sigma_2 + \ldots + a_{mn}\sigma_n = b_m, \end{cases}$$

где a_{ij}, b_i — неотрицательные целые числа. Требуется найти неотрицательные целочисленные решения этой системы. Это частный случай так называемой задачи целочисленного программирования. Рассмотрим для ее решения идеал

$$\langle y_j - x_1^{a_{1j}} x_2^{a_{2j}} \ldots x_m^{a_{nj}} \mid 1 \leqslant j \leqslant n \rangle$$

в кольце многочленов над $\mathbb{Q}$ и построим его базис Гребнера при исключающем упорядочении, где все переменные x_i старше любых переменных y_j. Пусть моном h — остаток монома $x_1^{b_1} x_2^{b_2} \ldots x_m^{b_m}$ при редукции относительно G. Утверждается, что если $h \notin \mathbb{Q}[y_1, \ldots, y_n]$, то неотрицательных целочисленных решений нет. Если же $h = y_1^{\sigma_1} y_2^{\sigma_2} \ldots y_n^{\sigma_n}$, то $(\sigma_1, \ldots, \sigma_n)$ — одно из решений. С помощью выбора подходящего упорядочения эта конструкция обобщается на случай, когда требуется найти решение, минимизирующее целевую функцию

$$\sum_{j=1}^{n} c_j \sigma_j.$$

Реализуйте алгоритм поиска решения задачи целочисленного программирования.

1. C. Wendler. *Groebner Bases with an Application to Integer Programming.* http://documents.kenyon.edu/math/CWendler.pdf

2. M. Kreuzer, L. Robbiano. *Computational Commutative Algebra (Part 1).* Tutorial 36. http://lib.mexmat.ru/books/1106

22. Регулярные последовательности. Пусть R — коммутативное кольцо. Последовательность элементов $a_1, \ldots, a_m \in R$ называется *регулярной*, если $a_1 \neq 0$ и образ a_k не является делителем нуля в $R/\langle a_1, \ldots, a_{k-1}\rangle$ для всех $k > 1$. Вам требуется написать программу, которая проверяла бы регулярность заданной последовательности в кольце многочленов $F[x_1, \ldots, x_n]$ с помощью вычислений подходящих базисов Гребнера.

Продемонстрируйте, что регулярность последовательности зависит, вообще говоря, от порядка элементов. Имеется ли такая зависимость в случае, когда элементы — однородные многочлены?

1. Д. Кокс, Дж. Литтл, Д. О'Ши, *Идеалы, многообразия и алгоритмы,* М., Мир, 2000. Глава 4. http://lib.mexmat.ru/books/1085, http://lib.mexmat.ru/books/36731

23. Прямая и обратная кинематические задачи. Пусть задан «робот» — конструкция из последовательно соединенных жестких сегментов. Один конец «руки» такого робота будет жестко закреплен, а другой будет «кистью». Сочленения могут быть шарнирными (рука вращается) или телескопическими (рука выдвигается). В качестве примера можно представить себе экскаватор.

Прямая кинематическая задача — дать явное описание положения кисти робота в зависимости от параметров сочленений (углов поворота, длины плеч). *Обратная кинематическая задача* — для данной точки найти такие параметры сочленений, при которых кисть робота окажется в этой точке.

Обе эти задачи могут быть переведены на язык алгебраических уравнений и исследованы с помощью базисов Гребнера. Напишите программу, решающую прямую и обратную кинематические задачи по заданным конфигурациям.

1. Д. Кокс, Дж. Литтл, Д. О'Ши, *Идеалы, многообразия и алгоритмы,* М., Мир, 2000. Глава 6. http://lib.mexmat.ru/books/1085, http://lib.mexmat.ru/books/36731

24. Уравнение эквидистанты. Пусть задано семейство алгебраических кривых (например, парабол $4py_0 - x_0^2 = 0$) и задан параметр $r > 0$. Задача — описать геометрическое место точек, равноудаленных от заданной кривой на расстояние r. Такие точки образуют *эквидистанту*. Обычно эквидистанту можно описать алгебраическим уравнением от x, y, r и параметров исходного семейства (в нашем примере — p). Напишите программу, которая бы по заданному семейству кривых вычисляла бы уравнение эквидистанты с помощью исключения неизвестных x_0, y_0.

25. Обобщение формулы Герона.

1. Пусть треугольник задан на плоскости координатами своих вершин. Понятно, что через эти координаты можно выразить квадраты сторон треугольника, а также его площадь. Исключая сами координаты из этой системы с помощью базиса Грёбнера, можно получить связь площади треугольника с его сторонами — формулу Герона. Проделайте это вычисление.
2. Выразите площадь треугольника через его медианы.
3. Пусть теперь тетраэдр задан в трехмерном пространстве координатами своих вершин. Получите с помощью базисов Гребнера формулу Эйлера, выражающую квадрат объема тетраэдра через квадраты длин его ребер.
4. Постарайтесь получить обобщение этой формулы на n-мерный симплекс (определитель Кэли-Менгера).

26. Гипотеза Casas–Alvero. Дан многочлен $f(x) \in F[x]$ над полем нулевой характеристики, обладающий следующими свойствами:

а) старший коэффициент $f(x)$ равен единице;

б) $f(x)$ имеет общие корни со всеми своими производными $f^{(k)}(x)$.

Тогда гипотеза утверждает, что $f(x)$ — степень линейного многочлена.

Проверьте эту гипотезу для небольших степеней f (2, 3, 4, ...). Для этого запишите f с неопределенными коэффициентами, выпишите систему уравнений на эти коэффициенты и исследуйте её с помощью базиса Грёбнера.