THINK FAST

Competitive Programming

From Problem 2 Solution in O(1)

Linear Algebra

Gaussian Elimination - 2

Mostafa Saad Ibrahim
PhD Student @ Simon Fraser University

Code

Gaussian Apps

- Solve system of linear Equations
- Solve system of linear Equations under Mod
- Compute Matrix Inverse
- Compute Determinant
- Some specific apps:
  - Patterns: Compute nth polynomial degree parameters
  - Solving Recurrence (useful if cyclic)

Gaussian and Prime Mod

- Solve following system under prime = 5
  - $x + y = 5 \ (\% \ 5)$
  - $3x + 6y = 1 \ (\% \ 5)$
  - Solution: $x = 3$ and $y = 2$
- Gaussian operations are finally +, -, *, /. So we can apply % (use Mod inverse)
- Better flip all input matrix to +ve values % p
- If Prime = 2. No division operation. Subtraction of 2 equations under mod 2 is just **xor** of values. So we can have faster code.

Code

Gaussian and Prime Mod

- As a note, solving
  - $x + y = 5 \ (\% \ 5)$
  - $3x + 6y = 1 \ (\% \ 5)$
- Same as solving
  - $x + y = 5 + 5\,k1$
  - $3x + 6y = 1 + 5\,k2$
- So don't be cheated with the extra 2 variables k1, k2...think like take all % 5

Gaussian and Matrix Inverse

$$\begin{pmatrix}1 & 2 & 1\\ 2 & 9 & 2\\ 1 & 7 & 2\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}$$

$$\begin{pmatrix}1 & 2 & 1\\ 0 & 5 & 0\\ 0 & 5 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ -2 & 1 & 0\\ -1 & 0 & 1\end{pmatrix}$$ $R_2=R_2-2*R_1$ AND $R_3=R_3-R_1$

$$\begin{pmatrix}1 & 2 & 1\\ 0 & 1 & 0\\ 0 & 5 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ -0.4 & 0.2 & 0\\ -1 & 0 & 1\end{pmatrix}$$ $R_2=R_2/5$

$$\begin{pmatrix}1 & 2 & 1\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ -0.4 & 0.2 & 0\\ 1 & -1 & 1\end{pmatrix}$$ $R_3=R_3-5*R_2$

$$\begin{pmatrix}1 & 2 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}0 & 1 & -1\\ -0.4 & 0.2 & 0\\ 1 & -1 & 1\end{pmatrix}$$ $R_1=R_1-R_3$

$$\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}0.8 & 0.6 & -1\\ -0.4 & 0.2 & 0\\ 1 & -1 & 1\end{pmatrix}$$ $R_1=R_1-2*R_2$

Determinant

- Recall...direct computations is expensive

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc.$$

$$\begin{aligned}\begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} &= a\begin{vmatrix} e & f \\ h & i \end{vmatrix} - b\begin{vmatrix} d & f \\ g & i \end{vmatrix} + c\begin{vmatrix} d & e \\ g & h \end{vmatrix} \\ &= a(ei - fh) - b(di - fg) + c(dh - eg) \\ &= aei + bfg + cdh - ceg - bdi - afh.\end{aligned}$$

Determinant and Elementary Operations

2	4
10	6

A = 2 * 6 - 4 * 10 = -28…
What if we swapped 2 rows?
What if we divided by factor?
What if we added 2 rows?

10	6
2	4

Swapping: A = 4 * 10 - 2 * 6= 28… Just multiply by -ve

1	2
10	6

Divide first row by 2
1 * 6 - 2*10 = -14 => Original Divided by 2

2	4
8	2

E2 -= E1
2 * 2 - 4 * 8 = -28 => NO effect

Gaussian and Determinant

- Given that the elementary row operations are valid and have controllable effect, gaussian algorithm can be **trivially changed**
- Swapping two rows multiplies the determinant by -1
- Multiplying a row by a nonzero scalar multiplies the determinant by the same scalar
- Adding to one row a scalar multiple of another does not change the determinant.

Gaussian and Determinant

- Given the division operation, computations will be saved in doubles...and may be we face precision problem
- Sol1: be very careful when casting |A| value
- Sol2: Use fractions instead of doubles, if problem says fraction values won't overflow
- Sol3: Use Big Integers/Big Decimals
- Sol4: Compute % prime if result won't over flow

Gaussian and Determinant

- **Compute % prime** if result won't over flow
- Assume we know final results fit in 32 bits.
- Find some problems p1 p2 ...pn such that
  - p1*p2...pn > MAX_ANSWER
  - p1*p2...pn < Data type limit (e.g. long long)
  - E.g. primes: 257, 263, 269, 271
- Compute determinant % Pi for every prime
- Use Chinese remainder theorem to compute final answer

Code

Gaussian and Patterns

- Assume we have some function:
- $f(n) = 5, 13, 24, 38, 55, 75, 98, \ldots$
  - You are informed that it has form:
  - $f(n) = an^2+bn+c$
  - Can you find $f(n)$?
- We have 3 unknowns..can we get 3 equations?
- Evaluate $f(1)$, $f(2)$, $f(3)$ => 3 equations
- Solve them:
  - $a = 3/2$, $b = 7/2$, $c = 0$
  - $f(n) = (3n^2+7n)/2$ starting with $n = 1$

Gaussian and Recurrence

- Recall Fibonacci?
  - F(n) = F(n-1) + F(n-2)   and F(0) = F(1) = 1
- If n = 4, we can think F(i) is variable / 5 vars
- Each F(i) is equation
  - E.g. F(4) = F(3) + F(2) =>   0 = F(3) + F(2) - F(4)
  - E.g. matrix row for F(4) can be: [-1 1 1 0 0 | 0]
- Now, just solve the system and compute F(4)
- As matrix is sparse, this is O(n^2)
- This is used in Dynamic Programming when recrrance are cyclic and can't be coded

Your Todo

- Implement O(n) solution for solving **Tridiagonal matrix** instead of Gaussian

$$\begin{bmatrix} b_1 & c_1 & & & 0 \\ a_2 & b_2 & c_2 & & \\ & a_3 & b_3 & \ddots & \\ & & \ddots & \ddots & c_{n-1} \\ 0 & & & a_n & b_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_n \end{bmatrix}.$$

تم بحمد الله

علمكم الله ما ينفعكم

ونفعكم بما تعلمتم

وزادكم علماً

Problems

- UVA 684, 10109, 10766, 10828, 11319, 11542, 11755
- HDU 4418 - Time travel, its solution
- http://codeforces.com/blog/entry/2536
- SPOJ(NWERC04H)