Phần IV

Câu 1:

a_{ij} với $1 \le i \le n, 1 \le j \le n$ và $b_1, b_2, ..., b_m$ lần lượt là các hệ số của các ẩn và kết quả(giá trị ở vế phải hay hệ số tự do) sau khi hệ phương trình được đưa về dạng bậc thang.

Gọi l là số phương trình trong hệ phương trình bậc thang có hệ số $k_i < n$

Khi đó, với hệ phương trình bậc thang , ta chứng minh được $l \le n$

Mặt khác, dựa vào điều kiện của hệ phương trình bậc thang, ta nhận thấy rằng ta chỉ nhận được tối đa n số nguyên $k_i < n$ phân biệt.

Với hệ có dạng bậc thang ta xét các trường hợp sau:

TH1: $l < n$

Nếu tồn tại i thỏa $\begin{cases} k_i \ge n \\ b_i \ne 0 \end{cases}$ => Hệ phương trình vô nghiệm

Ngược lại, nếu $\begin{cases} k_i \ge n \\ b_i = 0 \end{cases}$ với mọi i thì hệ phương trình có vô số nghiệm

TH2: $l = n$

Nếu tồn tại i thỏa $k_i \ge n$, ta xét b_i:

- Nếu $b_i \ne 0$ => Hệ phương trình vô nghiệm
- Nếu $b_i = 0$ với mọi i thỏa $k_i \ge n$ thì phương trình có duy nhất 1 nghiệm (Hệ phương trình bậc thang có n ẩn và n phương trình có chứa ẩn có hệ số khác 0)

Nếu $k_i < n$ với mọi i thì phương trình có duy nhất 1 nghiệm (Hệ phương trình bậc thang có n ẩn và n phương trình có chứa ẩn có hệ số khác 0)

TH3: $l > n$

Khi đó ta có được ít nhất l giá trị $k_i < n$ phân biệt (Vô Lý)

Câu 2:

Khi nhân ma trận A với:

- Ma trận $M(i;\lambda)$: Ta được một ma trận mới cùng kích thước có cá phần tử ở dòng i bị nhân gấp λ lần
- Ma trận $E(i;j)$: Ta nhận được một ma trận mới cùng kích thước có các phần tử ở dòng i đổi chỗ với các phần tử ở dòng j theo quy tắc A_{ik} đổi chỗ với A_{jk} với $k \in \{1;2;...;n\}$
- Ma trận $S(i,j;\lambda)$: Ta nhận được một ma trận mới cùng kích thước có các phần tử ở dòng A_{ik} bị biến đổi theo quy tắc $A_{ik} = A_{ik} + \lambda A_{jk}$ với $k \in \{1;2;...;n\}$. Hay có thể hiểu ta cộng các phần tử ở dòng i với λ lần các phần tử ở các cột tương ứng trên dòng j.

Khi nhân ma trận b với:

- Ma trận $M(i;\lambda)$: Ta được một ma trận mới cùng kích thước có cá phần tử ở dòng i bị nhân gấp λ lần
- Ma trận $E(i;j)$: Ta nhận được một ma trận mới cùng kích thước có phần tử ở dòng i đổi chỗ với phần tử ở dòng j hay có thể nói b_{i1} đổi chỗ với b_{j1}
- Ma trận $S(i,j;\lambda)$: Ta nhận được một ma trận mới cùng kích thước có phần tử ở dòng b_{i1} bị biến đổi theo quy tắc $b_{i1} = b_{i1} + \lambda b_{j1}$. Hay có thể hiểu ta cộng phần tử ở dòng i thêm với λ lần các phần tử ở trên dòng j.

Câu 3:

Như đã giải thích ở câu 2, việc nhân các ma trận sơ cấp vào 2 ma trận A và b chỉ tương tự với việc:

- Nhân một số khác không λ vào 2 vế của một phương trình: $M(i;\lambda)$
- Đổi chỗ 2 phương trình trong hệ phương trình: $E(i;j)$
- Cộng một một phương trình với λ (một số khác không) lần phương trình khác: $S(i,j;\lambda)$

Do đó, tập nghiệm của hệ phương trình $Ax=b$ và $TAx=Tb$ luôn trùng nhau với mọi ma trận sơ cấp $T \in R^{mxm}$

Câu 4:

1) Link code GitHub: Phần IV Câu 4
2) Thuật toán:

- Ta tìm hệ số k của từng phương trình trong hệ phương trình đã cho
- Ta tạo một list bỏ trống
- Tạo thêm 2 ma trận A0=A và B0=B
- Ta thực hiện vòng lặp sau n-1 lần:
  - Tìm phương trình i không nằm trong list bỏ có hệ số k nhỏ nhất và bé hơn n (số nghiệm của hệ phương trình)
  - Dùng phép khử Gauss để khử những hệ số ở cột k của các phương trình còn lại của ma trận A
  - Ở phép khử này, mỗi lần ta khử một phương trình t ta sẽ in ra môt ma trận trung gian S(t;j;lamda) với lamda là số đối của tỉ lệ giữa hệ số ở hàng t cột k với hệ số ở hàng i cột k.
  - Ta điều chỉnh hệ số ở mỗi hàng của ma trận b tương ứng
  - Thêm i vào list bỏ
  - Tính lại hệ số k của từng phương trình trừ những phương trình trong list bỏ
  - Ta cho A0=A,B0=B
  - Tiếp tục vòng lặp
- Ta nhận được một hệ phương trình với các phương trình trong nó có hệ số k đôi một khác nhau ngoại trừ những phương trình không chứa ẩn (hệ số n ẩn trong phương trình đó đều bằng 0)
- Ta tính lại các hệ số k và l của hệ phương trình mới này
- Ta tạo một list bỏ trống
- Ta xem những hàng ta chưa sắp xếp sẽ được gọi là hàng trống hay là chưa nằm trong list bỏ
- Tiếp tục sắp xếp lại hệ phương trình này thông qua vòng lặp:
  - Tìm phương trình i không nằm trong list bỏ có hệ số k nhỏ nhất và bé hơn n (số nghiệm của hệ phương trình)
  - Gọi hàng trống trên cùng ma trận là (hàng j)
  - Nếu i khác j:
  - Khi này ta in ra một ma trận trung gian M(i;j) để thể hiện việc đổi chỗ hàng i với hàng j (hàng trống trên cùng của ma trận)
  - Ta đổi chỗ các phần tử ở hàng i và hàng j ở các cột tương ứng trong cả ma trận A và b
  - Thêm j vào list bỏ

- Ta thu được một hệ phương trình có dạng bậc thang sau vòng lặp

Câu 5:

1) Link code github: [Phần IV Câu 5]
2) Thuật toán

- Thực chất câu 4 được làm dựa trên câu 5 nên thuật toán của 2 câu gần như tương tự chỉ khác ở câu 5, ta không in ra các ma trận trung gian trong quá trình giải mà chỉ in ra tập nghiệm của hệ phương trình.
  - Ta tìm hệ số k của từng phương trình trong hệ phương trình đã cho
  - Ta tạo một list bỏ trống
  - Tạo thêm 2 ma trận A0=A và B0=B
  - Ta thực hiện vòng lặp sau n-1 lần:
    - Tìm phương trình i không nằm trong list bỏ có hệ số k nhỏ nhất và bé hơn n (số nghiệm của hệ phương trình)
    - Dùng phép khử Gauss để khử những hệ số ở cột k của các phương trình còn lại của ma trận A
    - Ở phép khử này, mỗi lần ta khử một phương trình t ta sẽ cộng mỗi phần tử ở hàng t với lamda lần phần tử ở hàng j ở cột tương ứng (lamda là số đối của tỉ lệ giữa hệ số ở hàng t cột k với hệ số ở hàng i cột k).
    - Ta điều chỉnh hệ số ở mỗi hằng của ma trận A và b tương ứng
    - Thêm i vào list bỏ
    - Tính lại hệ số k của từng phương trình trừ những phương trình trong list bỏ
    - Ta cho A0=A, B0=B
    - Tiếp tục vòng lặp
  - Ta nhận được một hệ phương trình với các phương trình trong nó có hệ số k đôi một khác nhau ngoại trừ những phương trình không chứa ẩn (hệ số n ẩn trong phương trình đó đều bằng 0)
  - Ta tính lại các hệ số k và l của hệ phương trình mới này
  - Ta tạo một list bỏ trống
  - Ta xem những hàng ta chưa sắp xếp sẽ được gọi là hàng trống
  - Tiếp tục sắp xếp lại hệ phương trình này thông qua vòng lặp:
    - Tìm phương trình i không nằm trong list bỏ có hệ số k nhỏ nhất và bé hơn n (số nghiệm của hệ phương trình)
    - Gọi hàng trống trên cùng ma trận là (hàng j)

- Nếu i khác j:
- Khi này ta in ra một ma trận trung gian M(i;j) để thể hiện việc đổi chỗ hàng i với hàng j (hàng trống trên cùng của ma trận)
- Ta đổi chỗ các phần tử ở hàng i và hàng j ở các cột tương ứng trong cả ma trận A và b
- Thêm j vào list bỏ

- Ta thu được một hệ phương trình có dạng bậc thang sau vòng lặp
- Ta thu được một hệ phương trình có dạng bậc thang sau vòng lặp
- Ta tính lại các hệ số k và l của hệ phương trình mới này
- Ta tiếp tục biện luận nghiệm và giải như đã giải thích ở câu 1:
  - a_{ij} với $1 \le i \le n, 1 \le j \le n$ và $b_1, b_2, ..., b_m$ lần lượt là các hệ số của các ẩn và kết quả(giá trị ở vế phải hay hệ số tự do) sau khi hệ phương trình được đưa về dạng bậc thang.
  - Gọi l là số phương trình trong hệ phương trình bậc thang có hệ số $k_i < n$
  - Khi đó, với hệ phương trình bậc thang , ta chứng minh được $l \le n$
  - Mặt khác, dựa vào điều kiện của hệ phương trình bậc thang, ta nhận thấy rằng ta chỉ nhận được tối đa n số nguyên $k_i < n$ phân biệt.

Với hệ có dạng bậc thang ta xét các trường hợp sau:

TH1: $l < n$

Nếu tồn tại i thỏa $\begin{cases} k_i \ge n \\ b_i \ne 0 \end{cases}$ => Hệ phương trình vô nghiệm

Ngược lại, nếu $\begin{cases} k_i \ge n \\ b_i = 0 \end{cases}$ với mọi i thì hệ phương trình có vô số nghiệm

TH2: $l = n$

Nếu tồn tại i thỏa $k_i \ge n$, ta xét b_i:

- Nếu $b_i \ne 0$ => Hệ phương trình vô nghiệm
- Nếu $b_i = 0$ với mọi i thỏa $k_i \ge n$ thì phương trình có duy nhất 1 nghiệm (Hệ phương trình bậc thang có n ẩn và n phương trình có chứa ẩn có hệ số khác 0)

Nếu $k_i < n$ với mọi i thì phương trình có duy nhất 1 nghiệm (Hệ phương trình bậc thang có n ẩn và n phương trình có chứa ẩn có hệ số khác 0)

TH3: $l > n$

Khi đó ta có được ít nhất l giá trị $k_i < n$ phân biệt (Vô Lý)