

1. Tự học: Áp dụng thuật toán Tarjan tìm các BPLT của đồ thị có hướng (ví dụ)

Đồ thị gốc

Help Clear shift Delete Edit Undo Red Black

Áp dụng thuật toán Tarjan bằng cách duyệt đệ quy theo chiều sâu bắt đầu từ đỉnh 1

Lùi lại 1 bước Số bước: 52

SCC(1) -- --

1. Đánh số cho đỉnh 1 và đưa nó vào ngăn xếp S.
Gán $num[1] = min_num[1] = k = 1$; $k++$;
Đưa 1 vào S. Kết quả: S = 1
Thực hiện đánh số và đưa vào Stack ☒ 1

2. Với các đỉnh kề v của 1: 2,3 ☒ Xét ☒ 2

2a. v = '2' chưa duyệt ☒ Duyệt nó ☒ 3

SCC(2) -- --

1. Đánh số cho đỉnh 2 và đưa nó vào ngăn xếp S.
Gán $num[2] = min_num[2] = k = 2$; $k++$;
Đưa 2 vào S. Kết quả: S = 1,2
Thực hiện đánh số và đưa vào Stack ☒ 4

2. Với các đỉnh kề v của 2: 1,8 ☒ Xét ☒ 5

2b. v = '1' duyệt rồi nhưng vẫn còn trên stack ☒ Cập nhật $min_num[u]$ ☒ 6
Cập nhật $min_num[2] = \min(min_num[2], num[1]) = 1$ ☒ Cập nhật ☒ 7

2a. v = '8' chưa duyệt ☒ Duyệt nó ☒ 8

SCC(8) -- --

1. Đánh số cho đỉnh 8 và đưa nó vào ngăn xếp S.
Gán $num[8] = min_num[8] = k = 3$; $k++$;
Đưa 8 vào S. Kết quả: S = 1,2,8
Thực hiện đánh số và đưa vào Stack ☒ 9

2. Với các đỉnh kề v của 8: 7 ☒ Xét ☒ 10

2a. v = '7' chưa duyệt ☒ Duyệt nó ☒ 11

SCC(7) -- --

1. Đánh số cho đỉnh 7 và đưa nó vào ngăn xếp S.
Gán $num[7] = min_num[7] = k = 4$; $k++$;
Đưa 7 vào S. Kết quả: S = 1,2,8,7
Thực hiện đánh số và đưa vào Stack ☒ 12

2. Với các đỉnh kề v của 7: 6 ☒ Xét ☒ 13

2a. v = '6' chưa duyệt ☒ Duyệt nó ☒ 14

SCC(6) -- --

1. Đánh số cho đỉnh 6 và đưa nó vào ngăn xếp S.
Gán $num[6] = min_num[6] = k = 5$; $k++$;
Đưa 6 vào S. Kết quả: S = 1,2,8,7,6
Thực hiện đánh số và đưa vào Stack ☒ 15

2. Với các đỉnh kề v của 6: 7 ☒ Xét ☒ 16

2b. v = '7' duyệt rồi nhưng vẫn còn trên stack ☒ Cập nhật $min_num[u]$ ☒ 17

Cập nhật min_num[6] = min(min_num[6], num[7]) = 4

Cập nhật ✓ 18

3. Kiểm tra num và min_num của 6

num[6] == min_num[6]

num[6] != min_num[6] ✗ 19

Cập nhật min_num[7] = min(min_num[7], min_num[6]) = 4

Cập nhật ✓ 20

3. Kiểm tra num và min_num của 7

num[7] == min_num[7]

num[7] != min_num[7] ✓ 21

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 7

Các đỉnh được lấy ra: 6,7

Nội dung còn lại của ngăn xếp: 1,2,8

Cập nhật ✓ 22

Cập nhật min_num[8] = min(min_num[8], min_num[7]) = 3

Cập nhật ✓ 23

3. Kiểm tra num và min_num của 8

num[8] == min_num[8]

num[8] != min_num[8] ✓ 24

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 8

Các đỉnh được lấy ra: 8

Nội dung còn lại của ngăn xếp: 1,2

Cập nhật ✓ 25

Cập nhật min_num[2] = min(min_num[2], min_num[8]) = 1

Cập nhật ✓ 26

3. Kiểm tra num và min_num của 2

num[2] == min_num[2]

num[2] != min_num[2] ✗ 27

Cập nhật min_num[1] = min(min_num[1], min_num[2]) = 1

Cập nhật ✓ 28

2a. v = '3' chưa duyệt

Duyệt nó ✓ 29

SCC(3) -- --

1. Đánh số cho đỉnh 3 và đưa nó vào ngăn xếp S.

Gán num[3] = min_num[3] = k = 6; k++;

Đưa 3 vào S. Kết quả: S = 1,2,3

Thực hiện đánh số và đưa vào Stack ✓ 30

2. Với các đỉnh kề v của 3: 4,5

Xét ✓ 31

2a. v = '4' chưa duyệt

Duyệt nó ✓ 32

SCC(4) -- --

1. Đánh số cho đỉnh 4 và đưa nó vào ngăn xếp S.

Gán num[4] = min_num[4] = k = 7; k++;

Đưa 4 vào S. Kết quả: S = 1,2,3,4

Thực hiện đánh số và đưa vào Stack ✓ 33

2. Với các đỉnh kề v của 4: 2,7,8

Xét ✓ 34

2b. v = '2' duyệt rồi nhưng vẫn còn trên stack

Cập nhật min_num[u] ✓ 35

Cập nhật min_num[4] = min(min_num[4], num[2]) = 2

Cập nhật ✓ 36

2c. v = '7' duyệt rồi và không còn trên stack

Bỏ qua ✗ 37

2c. v = '8' duyệt rồi và không còn trên stack

Bỏ qua ✗ 38

3. Kiểm tra num và min_num của 4

num[4] == min_num[4]

num[4] != min_num[4] ✗ 39

Cập nhật min_num[3] = min(min_num[3], min_num[4]) = 2

Cập nhật ✓ 40

2a. v = '5' chưa duyệt ☒ 41

SCC(5)

1. Đánh số cho đỉnh 5 và đưa nó vào ngăn xếp S.
 Gán `num[5] = min_num[5] = k = 8` ; `k++`;
 Đưa 5 vào S. Kết quả: S = 1,2,3,4,5
 ☒ 42

2. Với các đỉnh kề v của 5: 3,7 ☒ 43

2b. v = '3' duyệt rồi nhưng vẫn còn trên stack ☒ 44
 Cập nhật `min_num[5] = min(min_num[5], num[3]) = 6` ☒ 45

2c. v = '7' duyệt rồi và không còn trên stack ☒ 46

3. Kiểm tra `num` và `min_num` của 5
 ☒ 47
 Cập nhật `min_num[3] = min(min_num[3], min_num[5]) = 2` ☒ 48

3. Kiểm tra `num` và `min_num` của 3
 ☒ 49
 Cập nhật `min_num[1] = min(min_num[1], min_num[3]) = 1` ☒ 50

3. Kiểm tra `num` và `min_num` của 1
 ☒ 51
 Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 1
 Các đỉnh được lấy ra: 5,4,3,2,1
 Nội dung còn lại của ngăn xếp:
 ☒ 52

Vẽ các thành phần liên thông

Help Clear shift Delete Edit Undo Red Black

2. Tự học: Áp dụng thuật toán Tarjan tìm các BPLT của đồ thị có hướng (ngẫu nhiên)
 *B1:

SCC(E)

1. Đánh số cho đỉnh E và đưa nó vào ngăn xếp S.

Gán $\text{num}[E] = \text{min_num}[E] = k = 6$; $k++$;

Đưa E vào S. Kết quả: S = C,F,B,A,D,E

Thực hiện đánh số và đưa vào Stack ☒ 16

2. Với các đỉnh kề v của E: Xét ☒ 17

3. Kiểm tra num và min_num của E

$\text{num}[E] == \text{min_num}[E]$ ☒ $\text{num}[E] != \text{min_num}[E]$ ☒ 18

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được E

Các đỉnh được lấy ra: E

Nội dung còn lại của ngăn xếp: C,F,B,A,D

Cập nhật ☒ 19

Cập nhật $\text{min_num}[D] = \min(\text{min_num}[D], \text{min_num}[E]) = 5$ Cập nhật ☒ 20

3. Kiểm tra num và min_num của D

$\text{num}[D] == \text{min_num}[D]$ ☒ $\text{num}[D] != \text{min_num}[D]$ ☒ 21

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được D

Các đỉnh được lấy ra: D

Nội dung còn lại của ngăn xếp: C,F,B,A

Cập nhật ☒ 22

Cập nhật $\text{min_num}[A] = \min(\text{min_num}[A], \text{min_num}[D]) = 4$ Cập nhật ☒ 23

2a. v = 'G' chưa duyệt ☒ Duyệt nó ☒ 24

SCC(G)

1. Đánh số cho đỉnh G và đưa nó vào ngăn xếp S.

Gán $\text{num}[G] = \text{min_num}[G] = k = 7$; $k++$;

Đưa G vào S. Kết quả: S = C,F,B,A,G

Thực hiện đánh số và đưa vào Stack ☒ 25

2. Với các đỉnh kề v của G: B Xét ☒ 26

2b. v = 'B' duyệt rồi nhưng vẫn còn trên stack Cập nhật $\text{min_num}[u]$ ☒ 27

Cập nhật $\text{min_num}[G] = \min(\text{min_num}[G], \text{num}[B]) = 3$ Cập nhật ☒ 28

3. Kiểm tra num và min_num của G

$\text{num}[G] == \text{min_num}[G]$ ☒ $\text{num}[G] != \text{min_num}[G]$ ☒ 29

Cập nhật $\text{min_num}[A] = \min(\text{min_num}[A], \text{min_num}[G]) = 3$ Cập nhật ☒ 30

3. Kiểm tra num và min_num của A

$\text{num}[A] == \text{min_num}[A]$ ☒ $\text{num}[A] != \text{min_num}[A]$ ☒ 31

Cập nhật $\text{min_num}[B] = \min(\text{min_num}[B], \text{min_num}[A]) = 3$ Cập nhật ☒ 32

3. Kiểm tra num và min_num của B

$\text{num}[B] == \text{min_num}[B]$ ☒ $\text{num}[B] != \text{min_num}[B]$ ☒ 33

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được B

Các đỉnh được lấy ra: G,A,B

Nội dung còn lại của ngăn xếp: C,F

Cập nhật ☒ 34

Cập nhật $\text{min_num}[F] = \min(\text{min_num}[F], \text{min_num}[B]) = 2$ Cập nhật ☒ 35

2c. v = 'D' duyệt rồi và không còn trên stack ☒ Bỏ qua ☒ 36

3. Kiểm tra num và min_num của F

$\text{num}[F] == \text{min_num}[F]$ ☒ $\text{num}[F] != \text{min_num}[F]$ ☒ 37

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được F

Các đỉnh được lấy ra: F

Nội dung còn lại của ngăn xếp: C

Cập nhật ☒ 38

Cập nhật $\text{min_num}[C] = \min(\text{min_num}[C], \text{min_num}[F]) = 1$ Cập nhật ☒ 39

2c. v = 'G' duyệt rồi và không còn trên stack ☐ Bỏ qua ☒ 40

3. Kiểm tra num và min_num của C

$\text{num}[C] == \text{min_num}[C]$ $\text{num}[C] != \text{min_num}[C]$ ☒ 41

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được C

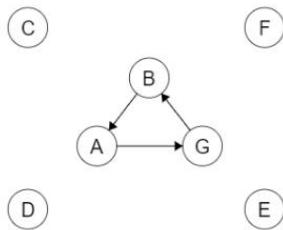
Các đỉnh được lấy ra: C

Nội dung còn lại của ngăn xếp:

Cập nhật ☒ 42

Vẽ các thành phần liên thông

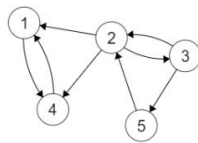
Help Clear shift Delete Edit Undo Red Black



*B2:

Đồ thị gốc

Help Clear shift Delete Edit Undo Red Black



Áp dụng thuật toán Tarjan bằng cách duyệt đệ quy theo chiều sâu bắt đầu từ đỉnh 2

in Lại lại 1 bước Số bước: 32

SCC(2) --- ---

1. Đánh số cho đỉnh 2 và đưa nó vào ngăn xếp S.

Gán $\text{num}[2] = \text{min_num}[2] = k = 1$; k++;

Đưa 2 vào S. Kết quả: S = 2

Thực hiện đánh số và đưa vào Stack ☒ 1

2. Với các đỉnh kề v của 2: 1,3,4

Xét ☒ 2

2a. v = '1' chưa duyệt Duyệt nó ☒ 3

SCC(1) --- ---

1. Đánh số cho đỉnh 1 và đưa nó vào ngăn xếp S.

Gán $\text{num}[1] = \text{min_num}[1] = k = 2$; k++;

Đưa 1 vào S. Kết quả: S = 2,1

Thực hiện đánh số và đưa vào Stack ☒ 4

2. Với các đỉnh kề v của 1: 4

Xét ☒ 5

2a. v = '4' chưa duyệt Duyệt nó ☒ 6

SCC(4) --- ---

1. Đánh số cho đỉnh 4 và đưa nó vào ngăn xếp S.

Gán num[4] = min_num[4] = k = 3; k++;

Đưa 4 vào S. Kết quả: S = 2,1,4

Thực hiện đánh số và đưa vào Stack 7

2. Với các đỉnh kề v của 4: 1

Xét 8

2b. v = '1' duyệt rồi nhưng vẫn còn trên stack Cập nhật min_num[u] 9

Cập nhật min_num[4] = min(min_num[4], num[1]) = 2 Cập nhật 10

3. Kiểm tra num và min_num của 4

num[4] == min_num[4] num[4] != min_num[4] X 11

Cập nhật min_num[1] = min(min_num[1], min_num(4)) = 2 Cập nhật 12

3. Kiểm tra num và min_num của 1

num[1] == min_num[1] num[1] != min_num[1] 13

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 1

Các đỉnh được lấy ra: 4,1

Nội dung còn lại của ngăn xếp: 2

Cập nhật 14

Cập nhật min_num[2] = min(min_num[2], min_num[1]) = 1 Cập nhật 15

2a. v = '3' chưa duyệt Duyệt nó 16

SCC(3) -- --

1. Đánh số cho đỉnh 3 và đưa nó vào ngăn xếp S.

Gán num[3] = min_num[3] = k = 4; k++;

Đưa 3 vào S. Kết quả: S = 2,3

Thực hiện đánh số và đưa vào Stack 17

2. Với các đỉnh kề v của 3: 2,5

Xét 18

2b. v = '2' duyệt rồi nhưng vẫn còn trên stack Cập nhật min_num[u] 19

Cập nhật min_num[3] = min(min_num[3], num[2]) = 1 Cập nhật 20

2a. v = '5' chưa duyệt Duyệt nó 21

SCC(5) -- --

1. Đánh số cho đỉnh 5 và đưa nó vào ngăn xếp S.

Gán num[5] = min_num[5] = k = 5; k++;

Đưa 5 vào S. Kết quả: S = 2,3,5

Thực hiện đánh số và đưa vào Stack 22

2. Với các đỉnh kề v của 5: 2

Xét 23

2b. v = '2' duyệt rồi nhưng vẫn còn trên stack Cập nhật min_num[u] 24

Cập nhật min_num[5] = min(min_num[5], num[2]) = 1 Cập nhật 25

3. Kiểm tra num và min_num của 5

num[5] == min_num[5] num[5] != min_num[5] X 26

Cập nhật min_num[3] = min(min_num[3], min_num[5]) = 1 Cập nhật 27

3. Kiểm tra num và min_num của 3

num[3] == min_num[3] num[3] != min_num[3] X 28

Cập nhật min_num[2] = min(min_num[2], min_num[3]) = 1 Cập nhật 29

2c. v = '4' duyệt rồi và không còn trên stack Bỏ qua X 30

3. Kiểm tra num và min_num của 2

num[2] == min_num[2] num[2] != min_num[2] 31

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 2

Các đỉnh được lấy ra: 5,3,2

Nội dung còn lại của ngăn xếp:

Cập nhật 32

Về các thành phần liên thông

Help Clear shift Delete Edit Undo Red Black

1

4

2

3

5

*B3:

Đồ thị gốc

Help

Clear

shift

Delete

Edit

Undo

Red

Black

Áp dụng thuật toán Tarjan bằng cách duyệt đệ quy theo chiều sâu bắt đầu từ đỉnh 1

Lỗi lại 1 bước

Số bước: 43

SCC(1)

--

--

1. Đánh số cho đỉnh 1 và đưa nó vào ngăn xếp S.

Gán $num[1] = min_num[1] = k = 1$; $k++$;

Đưa 1 vào S. Kết quả: S = 1

Thực hiện đánh số và đưa vào Stack

✓ 1

2. Với các đỉnh kề v của 1: 2,5

Xét

✓ 2

2a. v = '2' chưa duyệt

Duyệt nó

✓ 3

SCC(2)

--

--

1. Đánh số cho đỉnh 2 và đưa nó vào ngăn xếp S.

Gán $num[2] = min_num[2] = k = 2$; $k++$;

Đưa 2 vào S. Kết quả: S = 1,2

Thực hiện đánh số và đưa vào Stack

✓ 4

2. Với các đỉnh kề v của 2: 4

Xét

✓ 5

2a. v = '4' chưa duyệt

Duyệt nó

✓ 6

SCC(4)

--

--

1. Đánh số cho đỉnh 4 và đưa nó vào ngăn xếp S.

Gán $num[4] = min_num[4] = k = 3$; $k++$;

Đưa 4 vào S. Kết quả: S = 1,2,4

Thực hiện đánh số và đưa vào Stack

✓ 7

2. Với các đỉnh kề v của 4: 2,3

Xét

✓ 8

2b. v = '2' duyệt rồi nhưng vẫn còn trên stack

Cập nhật $min_num[u]$

✓ 9

Cập nhật $min_num[4] = \min(min_num[4], num[2]) = 2$

Cập nhật

✓ 10

2a. v = '3' chưa duyệt

Duyệt nó

✓ 11

SCC(3)

--

--

1. Đánh số cho đỉnh 3 và đưa nó vào ngăn xếp S.

Gán $num[3] = min_num[3] = k = 4$; $k++$;

Đưa 3 vào S. Kết quả: S = 1,2,4,3

Thực hiện đánh số và đưa vào Stack

✓ 12

2. Với các đỉnh kề v của 3:

Xét

✓ 13

3. Kiểm tra num và min_num của 3

$num[3] == min_num[3]$ $num[3] != min_num[3]$

✓ 14

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 3

Các đỉnh được lấy ra: 3

Nội dung còn lại của ngăn xếp: 1,2,4

Cập nhật

✓ 15

Cập nhật $min_num[4] = \min(min_num[4], min_num[3]) = 2$

Cập nhật

✓ 16

3. Kiểm tra num và min_num của 4

$num[4] == min_num[4]$ $num[4] != min_num[4]$

✗ 17

Cập nhật min_num[2] = min(min_num[2], min_num[4]) = 2

Cập nhật

18

3. Kiểm tra num và min_num của 2

num[2] == min_num[2]

num[2] != min_num[2]

19

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 2

Các đỉnh được lấy ra: 4,2

Nội dung còn lại của ngăn xếp: 1

Cập nhật

20

Cập nhật min_num[1] = min(min_num[1], min_num[2]) = 1

Cập nhật

21

2a. v = '5' chưa duyệt

Duyệt nó

22

SCC(5)

1. Đánh số cho đỉnh 5 và đưa nó vào ngăn xếp S.

Gán num[5] = min_num[5] = k = 5; k++;

Đưa 5 vào S. Kết quả: S = 1,5

Thực hiện đánh số và đưa vào Stack

23

2. Với các đỉnh kề v của 5: 7

Xét

24

2a. v = '7' chưa duyệt

Duyệt nó

25

SCC(7)

1. Đánh số cho đỉnh 7 và đưa nó vào ngăn xếp S.

Gán num[7] = min_num[7] = k = 6; k++;

Đưa 7 vào S. Kết quả: S = 1,5,7

Thực hiện đánh số và đưa vào Stack

26

2. Với các đỉnh kề v của 7: 3,6

Xét

27

2c. v = '3' duyệt rồi và không còn trên stack

Bỏ qua

28

2a. v = '6' chưa duyệt

Duyệt nó

29

SCC(6)

1. Đánh số cho đỉnh 6 và đưa nó vào ngăn xếp S.

Gán num[6] = min_num[6] = k = 7; k++;

Đưa 6 vào S. Kết quả: S = 1,5,7,6

Thực hiện đánh số và đưa vào Stack

30

2. Với các đỉnh kề v của 6: 4

Xét

31

2c. v = '4' duyệt rồi và không còn trên stack

Bỏ qua

32

3. Kiểm tra num và min_num của 6

num[6] == min_num[6]

num[6] != min_num[6]

33

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 6

Các đỉnh được lấy ra: 6

Nội dung còn lại của ngăn xếp: 1,5,7

Cập nhật

34

Cập nhật min_num[7] = min(min_num[7], min_num[6]) = 6

Cập nhật

35

3. Kiểm tra num và min_num của 7

num[7] == min_num[7]

num[7] != min_num[7]

36

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 7

Các đỉnh được lấy ra: 7

Nội dung còn lại của ngăn xếp: 1,5

Cập nhật

37

Cập nhật min_num[5] = min(min_num[5], min_num[7]) = 5

Cập nhật

38

3. Kiểm tra num và min_num của 5

num[5] == min_num[5]

num[5] != min_num[5]

39

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 5

Các đỉnh được lấy ra: 5

Nội dung còn lại của ngăn xếp: 1

Cập nhật ☒ 40

Cập nhật min_num[1] = min(min_num[1], min_num[5]) = 1 Cập nhật ☒ 41

3. Kiểm tra num và min_num của 1

num[1] == min_num[1] num[1] != min_num[1] ☒ 42

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 1

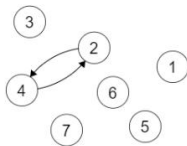
Các đỉnh được lấy ra: 1

Nội dung còn lại của ngăn xếp:

Cập nhật ☒ 43

Vẽ các thành phần liên thông

Help Clear shift Delete Edit Undo Red Black



*B4:

Đồ thị gốc

Help Clear shift Delete Edit Undo Red Black

Áp dụng thuật toán Tarjan bằng cách duyệt đệ quy theo chiều sâu bắt đầu từ đỉnh 5

Lùi lại 1 bước Số bước: 37

SCC(5) -- --

1. Đánh số cho đỉnh 5 và đưa nó vào ngăn xếp S.

Gán num[5] = min_num[5] = k = 1; k++;

Đưa 5 vào S. Kết quả: S = 5

Thực hiện đánh số và đưa vào Stack ☒ 1

2. Với các đỉnh kề v của 5: 2,4,6 Xét ☒ 2

2a. v = '2' chưa duyệt Duyệt nó ☒ 3

SCC(2) -- --

1. Đánh số cho đỉnh 2 và đưa nó vào ngăn xếp S.

Gán num[2] = min_num[2] = k = 2; k++;

Đưa 2 vào S. Kết quả: S = 5,2

Thực hiện đánh số và đưa vào Stack ☒ 4

2. Với các đỉnh kề v của 2: 4 Xét ☒ 5

2a. v = '4' chưa duyệt Duyệt nó ☒ 6

SCC(4) -- --

1. Đánh số cho đỉnh 4 và đưa nó vào ngăn xếp S.

Gán `num[4] = min_num[4] = k = 3`; `k++`;

Đưa 4 vào S. Kết quả: S = 5,2,4

Thực hiện đánh số và đưa vào Stack

7

2. Với các đỉnh kề v của 4: 3,5

Xét

8

2a. v = '3' chưa duyệt

Duyệt nó

9

SCC(3)

--

1. Đánh số cho đỉnh 3 và đưa nó vào ngăn xếp S.

Gán `num[3] = min_num[3] = k = 4`; `k++`;

Đưa 3 vào S. Kết quả: S = 5,2,4,3

Thực hiện đánh số và đưa vào Stack

10

2. Với các đỉnh kề v của 3: 2

Xét

11

2b. v = '2' duyệt rồi nhưng vẫn còn trên stack

Cập nhật `min_num[u]`

12

Cập nhật `min_num[3] = min(min_num[3], num[2]) = 2`

Cập nhật

13

3. Kiểm tra num và min_num của 3

`num[3] == min_num[3]`

`num[3] != min_num[3]`

14

Cập nhật `min_num[4] = min(min_num[4], min_num[3]) = 2`

Cập nhật

15

2b. v = '5' duyệt rồi nhưng vẫn còn trên stack

Cập nhật `min_num[u]`

16

Cập nhật `min_num[4] = min(min_num[4], num[5]) = 1`

Cập nhật

17

3. Kiểm tra num và min_num của 4

`num[4] == min_num[4]`

`num[4] != min_num[4]`

18

Cập nhật `min_num[2] = min(min_num[2], min_num[4]) = 1`

Cập nhật

19

3. Kiểm tra num và min_num của 2

`num[2] == min_num[2]`

`num[2] != min_num[2]`

20

Cập nhật `min_num[5] = min(min_num[5], min_num[2]) = 1`

Cập nhật

21

2b. v = '4' duyệt rồi nhưng vẫn còn trên stack

Cập nhật `min_num[u]`

22

Cập nhật `min_num[5] = min(min_num[5], num[4]) = 1`

Cập nhật

23

2a. v = '6' chưa duyệt

Duyệt nó

24

SCC(6)

--

1. Đánh số cho đỉnh 6 và đưa nó vào ngăn xếp S.

Gán `num[6] = min_num[6] = k = 5`; `k++`;

Đưa 6 vào S. Kết quả: S = 5,2,4,3,6

Thực hiện đánh số và đưa vào Stack

25

2. Với các đỉnh kề v của 6: 1

Xét

26

2a. v = '1' chưa duyệt

Duyệt nó

27

SCC(1)

--

1. Đánh số cho đỉnh 1 và đưa nó vào ngăn xếp S.

Gán `num[1] = min_num[1] = k = 6`; `k++`;

Đưa 1 vào S. Kết quả: S = 5,2,4,3,6,1

Thực hiện đánh số và đưa vào Stack

28

2. Với các đỉnh kề v của 1: 3

Xét

29

2b. v = '3' duyệt rồi nhưng vẫn còn trên stack

Cập nhật `min_num[u]`

30

Cập nhật `min_num[1] = min(min_num[1], num[3]) = 4`

Cập nhật

31

3. Kiểm tra num và min_num của 1

`num[1] == min_num[1]`

`num[1] != min_num[1]`

32

Cập nhật `min_num[6] = min(min_num[6], min_num[1]) = 4`

Cập nhật

33

3. Kiểm tra num và min_num của 6

`num[6] == min_num[6]`

`num[6] != min_num[6]`

34

Cập nhật min_num[5] = min(min_num[5], min_num[6]) = Cập nhật ✓ 35

3. Kiểm tra num và min_num của 5

✓ 36

Lấy các đỉnh trong ngăn xếp ra cho đến khi lấy được 5

Các đỉnh được lấy ra:

Nội dung còn lại của ngăn xếp:

Cập nhật ✓ 37

Vẽ các thành phần liên thông

Help Clear shift Delete Edit Undo Red Black

