DFS

56
Trần Nguyễn Nhật Huy
B2113333

Khởi tạo stack S;
tạo rỗng : S. top_idx = -1;
Đưa 1 vào S:
   S. top_idx = 0;
   S. data[0] = 1;
S = {1};

Lần lặp thứ 1:

while (S. top_idx = 0 != -1) {
   int u = S. data[0] = 1;
   S. top_idx = -1, S = { }
   if (mark[1] = 0 == 0) → bỏ qua điều kiện

In " Duyệt 1\n" ra màn hình

mark[1] = 1;
for {v là các đỉnh trong đồ thị) {
   v = 1 → bỏ qua điều kiện
   v = 2 → bỏ qua điều kiện
   v = 3 → bỏ qua điều kiện
   v = 4 → thoả điều kiện adjacent(pG, 1, 4) = 1
Đưa 4 vào S:
   S. top_idx = 0;
   S. data[0] = 4;
S = {4};
   v = 5 → thoả điều kiện adjacent(pG, 1, 5) = 1
Đưa 5 vào S:
   S. top_idx = 1;
   S. data[1] = 5; S = {4, 5}
   v = 6 → bỏ qua điều kiện
   v = 7 → bỏ qua điều kiện }

Lần lặp thứ 2

S = {4, 5}
while (S. top_idx = 1 != -1) {
   int u = S. data[1] = 5
   S. top_idx = 0, S = {4}
   mark[5] = 0 → bỏ qua điều kiện.
In " Duyệt 5\n" ra màn hình
mark[5] = 1
for (v là các đỉnh trong đồ thị) {
   v = 2, 3, 6, 7, 5 → bỏ qua điều kiện.
   v = 1 → thoả điều kiện adjacent(pG, 5, 1) = 1
Đưa 1 vào S:
   S. top_idx = 1
   S. data[1] = 1.
   S = {4, 1}
   v = 4 → thoả điều kiện adjacent(pG, 5, 4) = 1
Đưa 4 vào S:
   S. top_idx = 2
   S. data[2] = 4.
   S = {4, 1, 4}.
   v = 8 → thoả điều kiện adjacent(pG, 5, 8) = 1
Đưa 8 vào S:
   S. top_idx = 3
   S. data[3] = 8.
   S = {4, 1, 4, 8}.   }

Lần lặp thứ 3

S = {4, 1, 4, 8}
while (S. top_idx = 3 != -1) {
   int u = S. data[3] = 8
   S. top_idx = 2, S = {4, 1, 4}
   mark[8] = 0 → bỏ qua điều kiện
In " Duyệt 8\n" ra màn hình
mark[8] = 1
for (v là các đỉnh trong đồ thị) {
   v = 1, 3, 6, 7, 8 → bỏ qua điều kiện.
   v = 2 → thoả adjacent(pG, 8, 2) = 1.
Đưa 2 vào S
   S. top_idx = 3
   S. data[3] = 2, S = {4, 1, 4, 2}
   v = 4 → thoả adjacent(pG, 8, 4) = 1

①

Đưa 4 vào S:
   S. top_idx = 4.
   S. data[4] = 4, S = {4,1,4,2,4}
②. v = 5 thoả adjacent(pG, 8,5) = 1
   Đưa 5 vào S
      S. top_idx = 5
      S. data[5] = 5, S = {4,1,4,2,4,5}.
   }
}

[Lần lặp thứ 4]

S = {4,1,4,2,4,5}
while (S. top_idx = 5 != -1) {
   int u = S. data[5] = 5
   S. top_idx = 4., S = {4,1,4,2,4}
   mark[5] = 1 thoả mark[5] != 0
      Chuyển qua vòng lặp kế tiếp.

[Lần lặp thứ 5]

S = {4,1,4,2,4}
while (S. top_idx = 4 != -1) {
   int u = S. data[4] = 4
   S. top_idx = 3, S = {4,1,4,2}
   mark[4] = 0 → bỏ qua điều kiện.
   In " Duyệt 4 \n " ra màn hình.
   mark[4] = 1
   for(v là các đỉnh trong đồ thị) {
      v = 3,6,7,4 → bỏ qua
      v = 1 → thoả adjacent (pG, 4,1) = 1
      Đưa 1 vào S:
         S. top_idx = 4
         S. data[4] = 1, S = {4,1,4,2,1}
      Đưa 2 vào S:
         S. top_idx = 5
         S. data[5] = 2, S = {4,1,4,2,1,2}
      Đưa 5 vào S:
         S. top_idx = 6
         S. data[6] = 5, S = {4,1,4,2,1,2,5}
      Đưa 8 vào S:
         S. top_idx = 7.
         S. data[7] = 8, S = {4,1,4,2,1,2,5,8}

   }
}

[Lần lặp thứ 6]

S = {4,1,4,2,1,2,5,8}
while (S. top_idx = 7 != -1) {
   int u = S. data[7] = 8.
   S. top_idx = 6, S = {4,1,4,2,1,2,5}.
   mark[8] = 1 thoả mark[8] != 0
      chuyển qua vòng lặp kế tiếp.

[Lần lặp thứ 7]

S = {4,1,4,2,1,2,5}
while (S. top_idx = 6 != -1) {
   int u = S. data[6] = 5
   S. top_idx = 5, S = {4,1,4,2,1,2}.
   mark[5] = 1 thoả mark[5] != 0
      chuyển qua vòng lặp kế tiếp.

[Lần lặp thứ 8]

S = {4,1,4,2,1,2}
while (S. top_idx = 5 != -1) {
   int u = S. data[5] = 2.
   S. top_idx = 4, S = {4,1,4,2,1}
   mark[2] = 0 → bỏ qua điều kiện
   In " Duyệt 2 \n " ra màn hình
   mark[2] = 1.
   for(v là các đỉnh trong đồ thị) {
      v = 1,5,2 → bỏ qua
      v = 3 → thoả adjacent (pG, 2,3) = 1, đưa 3 vào S:
         S. top_idx = 5
         S. data[5] = 3, S = {4,1,4,2,1,3}
      v = 4 → thoả adjacent (pG, 2,4) = 1, đưa 4 vào S:
         S. top_idx = 6
         S. data[6] = 4, S = {4,1,4,2,1,3,4}
      v = 6 → thoả adjacent (pG, 2,6) = 1, đưa 6 vào S
         S. top_idx = 7.
         S. data[7] = 6, S = {4,1,4,2,1,3,4,6}
      v = 7 → thoả adjacent (pG, 2,7) = 1, đưa 7 vào S
         S. top_idx = 8
         S. data[8] = 7, S = {4,1,4,2,1,3,4,6,7} ②

V = 8 → thoả adjacent (pG, 2,8)=1, đưa 8 vào S
S. top-idk = 9
S. data [9] = 8, S = {4,1,4,2, 1,3,4,6,7,8}

Lần lặp thứ 9

S = {4,1,4,2,1,3,4,6,7,8}

while (S.top-idx = 9 != -1){

    int u = S. data [9] = 8

    S.top-idx = 8 S = {4,1,4,2,1,3,4,6,7}

    mark [8] = 1 → thoả mark[8] != 0

    chuyển qua vòng lặp kế tiếp

Lần lặp thứ 10

S = {4,1,4,2,1,3,4,6,7}

while (S. top- idx = 8 != -1){

    int u = S. data [8] = 7.

    S.top-idx = 7, S = {4,1,4,2,1,3,4,6}

    mark[7] = 0 → bỏ qua điều kiện

    In " Duyệt 7\n" ra màn hình

    mark [7] = 1

    for(v là các đỉnh trong đồ thị){

    v = 1,4,5,6,7,8 → bỏ qua

    v = 2 → thoả adjacent (pG, 7,2) = 1, đưa 2 vào S

        S.top-idx = 8

        S. data [8] = 2, S = {4,1,4,2,1,3,4,6,2}

    v = 3 → thoả adjacent (pG, 7,3)=1, đưa 3 vào S

        S. top- idx = 9

        S. data [9] = 3, S = {4,1,4,2,1,3,4,6,2,3}

Lần lặp thứ 11

S = {4,1,4,2,1,3,4,6,2,3}

while (S top-idx = 9 != -1){

    int u = S. data [9] = 3

    S.top-idx = 8, S = {4,1,4,2,1,3,4,6,2}

    mark[3] = 0 → bỏ qua điều kiện

    In " Duyệt 3\n" ra màn hình

    mark [3] = 1

    for(v là các đỉnh trong đồ thị)

    v = 1,3,4,5,8 → bỏ qua

v = 2 → thoả adjacent (pG, 3,2) = 1, đưa 2 vào S.
    S. top-idx = 9
    S. data[9] = 2, S = {4,1,4,2,1,3,4,6,2,2}
v = 6 → thoả adjacent (pG, 3,6) = 1, đưa 3 vào S.
    S. top- idx = 10
    S. data [10] = 6, S = {4,1,4,2,1,3,4,6,2,2,6}
v = 7 → thoả adjacent (pG, 3,7) = 1, đưa 7 vào S.
    S.top- idx = 11
    S. data [11] = 7, S = {4,1,4,2,1,3,4,6,2,2,6,7}

Lần lặp thứ 12

S = {4,1,4,2,1,3,4,6,2,2,6,7}

while (S. top-idx = 11 != -1){

    int u = S. data [11] = 7.

    S.top- idx = 10, S = {4,1,4,2,1,3,4,6,2,2,6}

    mark[7] = 1 → thoả mark[7] != 0

    chuyển qua vòng lặp kế tiếp.

Lần lặp thứ 13

S = {4,1,4,2,1,3,4,6,2,2,6}

while (S.top-idx = 10 != -1)

    int u = S. data [10] = 6.

    S.top- idx = 9, S = {4,1,4,2,1,3,4,6,2,2}

    mark[6] = 0 → bỏ qua điều kiện

    In " Duyệt 6\n" ra màn hình

    mark [6] = 1

    for(v là các đỉnh trong đồ thị){

    v = 1,4,5,6,7,8 → bỏ qua

    v = 2 → thoả adjacent (pG, 6,2) = 1, đưa 8 vào S.

        S.top-idx = 10

        S. data [10] = 2, S = {4,1,4,2,1,3,4,6,2,2,2}

    v = 3 → thoả adjacent (pG, 6,3) = 1, đưa 3 vào S.

        S. top- idx = 11

        S. data [11] = 3, S = {4,1,4,2,1,3,4,6,2,2,2,3}

Lần lặp thứ 14

S = {4,1,4,2,1,3,4,6,2,2,2,3}

while (S. top-idx = 11 != -1){

    int u = S. data [11] = 3.

    S. top-idx = 10, S = {4,1,4,2,1,3,4,6,2,2,2}

    mark [3] = 1 → thoả mark [3] != 0 → chuyển qua lặp kế tiếp

③

Lần thứ 15

$S = \{4, 1, 4, 2, 1, 3, 4, 6, 2, 2, 2\}$

while (S.top_idx = 10 != -1) {

   int u = S.data[10] = 2.

   S.top_idx = 9, $S = \{4, 1, 4, 2, 1, 3, 4, 6, 2, 2\}$

   mark[2] = 1 -> thoái mark[2] = 1

   chuyển qua vòng lặp kế tiếp.

Lần thứ 16

$S = \{4, 1, 4, 2, 1, 3, 4, 6, 2, 2\}$

while (S.top_idx = 9 != -1) {

   int u = S.data[9] = 2.

   S.top_idx = 8, $S = \{4, 1, 4, 2, 1, 3, 4, 6, 2\}$

   mark[2] = 1 -> chuyển qua vòng lặp kế tiếp.

Lần thứ 17

$S = \{4, 1, 4, 2, 1, 3, 4, 6, 2\}$

while (S.top_idx = 8 != -1) {

   int u = S.data[8] = 2.

   S.top_idx = 7, $S = \{4, 1, 4, 2, 1, 3, 4, 6\}$

   mark[2] = 1 -> chuyển qua vòng lặp kế tiếp.

Lần thứ 18

$S = \{4, 1, 4, 2, 1, 3, 4, 6\}$

while (S.top_idx = 7 != -1) {

   int u = S.data[7] = 6.

   S.top_idx = 6, $S = \{4, 1, 4, 2, 1, 3, 4\}$

   mark[6] = 1 -> chuyển qua vòng lặp kế tiếp.

Lần thứ 19

$S = \{4, 1, 4, 2, 1, 3, 4\}$

while (S.top_idx = 6 != -1) {

   int u = S.data[6] = 4.

   S.top_idx = 5, $S = \{4, 1, 4, 2, 1, 3\}$

   mark[4] = 1 -> chuyển qua vòng lặp kế tiếp.

Lần thứ 20

$S = \{4, 1, 4, 2, 1, 3\}$

while (S.top_idx = 5 != -1) {

   int u = S.data[5] = 3.

   S.top_idx = 4, $S = \{4, 1, 4, 2, 1\}$

   mark[3] = 1 -> chuyển qua vòng lặp kế tiếp.

Lần thứ 21

$S = \{4, 1, 4, 2, 1\}$

while (top_idx = 4 != -1) {

   int u = S.data[4] = 1.

   S.top_idx = 3, $S = \{4, 1, 4, 2\}$

   mark[1] = 1 -> chuyển sang vòng lặp kế tiếp

Lần thứ 22

$S = \{4, 1, 4, 2\}$

while (S.top_idx = 3 != -1) {

   int u = S.data[3] = 2.

   S.top_idx = 2, $S = \{4, 1, 4\}$

   mark[2] = 1 -> chuyển sang vòng lặp kế tiếp.

Lần thứ 23

$S = \{4, 1, 4\}$

while (S.top_idx = 2 != -1) {

   int u = S.data[2] = 4

   S.top_idx = 1, $S = \{4, 1\}$

   mark[4] = 1 -> chuyển sang vòng lặp kế tiếp.

Lần thứ 24

$S = \{4, 1\}$

while (S.top_idx = 1 != -1) {

   int u = S.data[1] = 1

   S.top_idx = 0, $S = \{4\}$

   mark[1] = 1 -> chuyển sang vòng lặp kế tiếp.

Lần thứ 25

$S = \{4\}$

while (S.top_idx = 0 != -1) {

   int u = S.data[0] = 4.

   S.top_idx = -1, $S = \{ \}$

   mark[4] = 1 -> chuyển sang vòng lặp kế tiếp

Lần thứ 26

$S = \{ \}$

while (S.top_idx = -1 == -1) {

   kết thúc vòng lặp

}

④

BFS
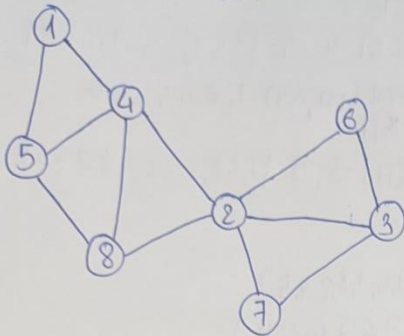
Khai báo hàng đợi Q
tạo rỗng : Q.font = 0; Q.rear = -1

đưa 1 vào Q.
   Q.rear = 0
   Q.data[0] = 1

Q = {1}

Lần lặp 1
   while (Q.front ⩽ Q.rear)(0 ⩽ 0) {
      u = Q.data[0] = 1
      Q.front = 1 , Q = { }
      mark[1] = 0 → bỏ qua
      In " Duyệt 1\n".
      mark[1] = 1.
      for (v : các đỉnh kề trong đồ thị) {
         v = 1,2,3,6,7,8 → bỏ qua
         v = 4 → adjacent (pG,1,4) =1, đưa 4 vào Q:
            Q.rear = 1
            Q.data[1] = 4, Q = {4}.
         v = 5 → adjacent(pG, 1,5) = 1, đưa 5 vào Q:
            Q.rear = 2.
            Q.data[2] = 5, Q = {4,5}

Lần lặp 2
   Q = {4,5}
   while (Q.front = 1 ⩽ Q.rear = 2) {
      u = Q.data[1] = 4.
      Q.front = 2, Q = {5}.
      mark[4] = 0 → bỏ qua.
      In "Duyệt 4\n".
      mark[4] = 1
      for(v: các đỉnh trong đồ thị) {

---

         v = 3,4,6,7 → bỏ qua
         v = 1 → adjacent (pG,4,1) = 1, đưa 1 vào Q :
            Q.rear = 3
            Q.data[3] = 1. , Q = {5,1}
         v = 2 → adjacent (pG, 4,2)=1, đưa 2 vào Q :
            Q.rear = 4
            Q.data[4] = 2, Q = {5,1,2}
         v = 5 → adjacent (pG,4,5) = 1, đưa 5 vào Q:
            Q.rear = 5
            Q.data[5] = 5, Q = {5,1,2,5}
         v = 8 → adjacent(pG, 4,8) = 1, đưa 8 vào Q:
            Q.rear = 6
            Q.data[6] = 8, Q = {5,1,2,5,8}

Lần lặp 3
   Q = {5,1,2,5,8}
   while (Q.front = 2 ⩽ Q.rear = 6) {
      u = Q.data[2] = 5
      Q.front = 3, Q = {1,2,5,8}
      mark[5] = 0 → bỏ qua
      In " Duyệt 5\n"
      mark[5] = 1
      for (v : các đỉnh trong đồ thị) {
         v = 2,3,5,6,7 → bỏ qua.
         v = 1 → adjacent (pG,5,1)=1, đưa 1 vào Q.
            Q.rear = 7.
            Q.data[7] = 1, Q = {1,2,5,8,1}
         v = 4 → adjacent (pG,5,4) = 1, đưa 4 vào Q.
            Q.rear = 8
            Q.data[8] = 1, Q = {1,2,5,8,1,4}
         v = 8 → adjacent (pG,5,8) = 1, đưa 8 vào Q
            Q.rear = 9
            Q.data[9] = 8, Q = {1,2,5,8,1,4,8}

Lần lặp 4
   Q = {1,2,5,8,1,4,8}
   while (Q.front = 3 ⩽ Q.rear = 9) {
      u = Q.data[3] = 1.
      Q.front = 4, Q = {2,5,8,1,4,8}
      mark[1] = 1 → chuyển sang vòng lặp kế tiếp.

Lần lặp 5
   Q = {2,5,8,1,4,8}
   while (Q.front = 4 ⩽ Q.rear = 9) {

①

u = Q. data [4] = 2.
Q. front = 5 , Q = { 5,8,1,4,8 }
mark [2] = 0 -> bỏ qua
In " Duyệt 2\n"
mark [2] = 1
for (v: các đỉnh trên đồ thị) {
  v = 1,2,5 -> bỏ qua
  v = 3 -> adjacent (pG, 2,3) = 1, đưa 3 vào S.
    Q. rear = 10
    Q. data [10] = 3, Q = { 5,8,1,4,8,3 }
  v = 4 -> adjacent (pG, 2,4) = 1, đưa 4 vào S.
    Q. rear = 11
    Q. data [11] = 4, Q = { 5,8,1,4,8,3 ,4 }
  v = 6 -> adjacent (pG, 2,6) = 1, đưa 2 vào S.
    Q. rear = 12
    Q. data [12] = 6, Q = { 5,8,1,4,8,3,4,6 }
  v = 7 -> adjacent (pG, 2,7) = 1, đưa 7 vào S.
    Q. rear = 13
    Q. data [13] = 7, Q = { 5,8,1,4,8,3,4,6,7 }
  v = 8 -> adjacent (pG, 2, 8) = 1, đưa 8 vào S.
    Q. rear = 14
    Q. data [14] = 8, Q = { 5,8,1,4,8,3,4,6,7,8 }

Lần lặp 6

Q = { 5,8,1,4,8,3,4,6,7,8 }
while (front = 5 ≤ Q. rear = 14) {
  u = Q. data [5] = 5
  Q. front = 6, Q = { 8,1,4,8,3,4,6,7,8 }
  mark [5] = 1 -> chuyển sang vòng lặp kế tiếp.

Lần lặp 7

Q = { 8,1,4,8,3,4,6,7,8 }
while (Q. front = 6 ≤ Q. rear = 14) {
  u = Q. data [6] = 8
  Q. front = 7, Q = { 1,4,8,3,4,6,7,8 }
  mark [8] = 0 -> bỏ qua
  In " Duyệt 8\n"
  mark [8] = 1
  for (v: các đỉnh kề trên đồ thị) {
    v = 1,3,6,7,8 -> bỏ qua.
    v = 2 -> adjacent (pG, 8,2) = 1, đưa 2 vào Q
      Q. rear = 15
      Q. data [15] = 2, Q = { 1,4,8,3,4,6,7,8,2 }

v = 4 -> adjacent (pG, 8, 4) = 1, đưa 4 vào Q:
  Q. rear = 16.
  Q. data [16] = 4, Q = { 1,4,8,3,4,6,7,8,2,4 }
v = 5 -> adjacent (pG, 8,5) = 1, đưa 5 vào Q:
  Q. rear = 17.
  Q. data [17] = 5, Q = { 1,4,8,3,4,6,7,8,2,4,5 }

Lần lặp 8

Q = { 1,4,8,3,4,6,7,8,2,4,5 }
while (Q. front = 7 ≤ Q. rear = 17) {
  u = Q. data [7] = 1.
  Q. front = 8, Q = { 4,8,3,4,6,7,8,2,4,5 }
  mark [1] = 1 -> chuyển sang vòng lặp kế.

Lần lặp 9

Q = { 4,8,3,4,6,7,8,2,4,5 }
while (Q. front = 8 ≤ Q. rear = 17) {
  u = Q. data [8] = 4
  Q. front = 9, Q = { 8,3,4,6,7,8,2,4,5 }
  mark [4] = 1 -> chuyển sang vòng lặp kế tiếp

Lần lặp 10

Q = { 8,3,4,6,7,8,2,4,5 }
while (Q. front = 9 ≤ Q. rear = 17).
  u = Q. data [9] = 8
  Q. front = 10, Q = { 3,4,6,7,8,2,4,5 }
  mark [8] = 1 -> chuyển sang vòng lặp kế tiếp.

Lần lặp 11

Q = { 3,4,6,7,8,2,4,5 }
while (Q. front = 10 ≤ Q. rear = 17) {
  u = Q. data [10] = 3
  Q. front = 11, Q = { 4,6,7,8,2,4,5 }
  mark [3] = 0 -> bỏ qua
  In " Duyệt 3\n"
  mark [3] = 1
  for (v: các đỉnh trên đồ thị)
    v = 1,3,4,5,8 -> bỏ qua.
    v = 2 -> adjacent (pG, 3,2) = 1, đưa 2 vào Q:
      Q. rear = 18
      Q. data [18] = 2, Q = { 4,6,7,8,2,4,5,2 }
    v = 8 -> adjacent (pG, 3,6) = 1, đưa 6 vào Q:
      Q. rear = 19
      Q. data [19] = 6, Q = { 4,6,7,8,2,4,5,2,6 } ②
    v = 7 -> adjacent (pG, 3,7) = 1, đưa 7 vào Q:
      Q. rear = 20
      Q. data [20] = 7, Q = { 4,6,7,8,2,4,5,2,6,7 }

Lần lặp 12

$Q = \{4,6,7,8,2,4,5,2,6,7\}$

while (Q.front = 11 < Q.rear = 20) {

    u = Q.data[11] = 4.

    Q.front = 12, Q = {6,7,8,2,4,5,2,6,7}

    mark[4] = 1 → chuyển sang vòng lặp kế

Lần lặp 13

$Q = \{6,7,8,2,4,5,2,6,7\}$

while (Q.front = 12 < Q.rear = 20) {

    u = Q.data[12] = 6.

    Q.front = 13, Q = {7,8,2,4,5,2,6,7}

    mark[6] = 0 → bỏ qua.

In "Duyệt 6\n"

mark[6] = 1.

for (v: các đỉnh trên đồ thị) {

    v = 1,4,5,7,6,8 → bỏ qua

    v = 2 → adjacent (pG,6,2) = 1, đưa 2 vào Q:

        Q.rear = 21

        Q.data[21] = 2, Q = {7,8,2,4,5,2,6,7,2}

    v = 3 → adjacent (pG,6,3) = 1, đưa 3 vào Q:

        Q.rear = 22

        Q.data[22] = 3, Q = {7,8,2,4,5,2,6,7,2,3}

Lần lặp 14

$Q = \{7,8,2,4,5,2,6,7,2,3\}$

while (Q.front = 13 < Q.rear = 22) {

    u = Q.data[13] = 7.

    Q.front = 14, Q = {8,2,4,5,2,6,7,2,3}.

    mark[7] = 0 → bỏ qua

In "Duyệt 7\n"

mark[7] = 1

for (v: các đỉnh trên đồ thị) {

    v = 1,4,5,6,7,8 → bỏ qua

    v = 2 → adjacent (pG,7,2) = 1, đưa 2 vào Q:

        Q.rear = 23

        Q.data[23] = 2, Q = {8,2,4,5,2,6,7,2,3,2}

    v = 3 → adjacent (pG,7,3) = 1, đưa 3 vào Q:

        Q.rear = 24

        Q.data[24] = 3; Q = {8,2,4,5,2,6,7,2,3,2,3}

Lần lặp 15

$Q = \{8,2,4,5,2,6,7,2,3,2,3\}$

while (Q.front = 14 < Q.rear = 24) {

    u = Q.data[14] = 8.

    Q.front = 15, Q = {2,4,5,2,6,7,2,3,2,3}

    mark[8] = 1 → chuyển sang vòng lặp kế.

Lần lặp 16

$Q = \{2,4,5,2,6,7,2,3,2,3\}$

while (Q.front = 15 < Q.rear = 24) {

    u = Q.data[15] = 2.

    Q.front = 16, Q = {4,5,2,6,7,2,3,2,3}

    mark[2] = 1 → chuyển sang vòng lặp kế.

Lần lặp 17

$Q = \{4,5,2,6,7,2,3,2,3\}$

while (Q.front = 16 < Q.rear = 24 }

    u = Q.data[16] = 4.

    Q.front = 17, Q = {5,2,6,7,2,3,2,3}

    mark[4] = 1 → chuyển sang vòng lặp kế.

Lần lặp 18

$Q = \{5,2,6,7,2,3,2,3\}$

while (Q.front = 17 < Q.rear = 24 }

    u = Q.data[17] = 5.

    Q.front = 18, Q = {2,6,7,2,3,2,3}

    mark[5] = 1 → chuyển sang vòng lặp mới.

Lần lặp 19

$Q = \{2,6,7,2,3,2,3\}$

while (Q.front = 18 < Q.rear = 24) {

    u = Q.data[18] = 2

    Q.front = 19, Q = {6,7,2,3,2,3}

    mark[2] = 1 → chuyển sang vòng lặp mới.

Lần lặp 20

$Q = \{6,7,2,3,2,3\}$

while (Q.front = 19 < Q.rear = 24) {

    u = Q.data[19] = 6.

    Q.front = 20, Q = {7,2,3,2,3}

    mark[6] = 1 → chuyển sang vòng lặp kế.

③

Lần lặp 21

Q = {7, 2, 3, 2, 3}

while (Q. front = 20 ≤ Q. rear = 24)}

    u = Q. data [20] = 7

    Q. front = 21, Q = {2, 3, 2, 3}

    mark [7] = 1 →) vòng lặp lẻ.

Lần lặp 22

Q = { 2, 3, 2, 3}

while (Q. front = 21 ≤ Q. rear = 24}.

    u = Q. data [21] = 2.

    Q. front = 22, Q = {3, 2, 3}

    mark [2] = 1 →) vòng lặp lẻ.

Lần lặp 23

Q = {3, 2, 3}

while (Q. front = 22 ≤ Q. rear = 24)}.

    u = Q. data [22] = 3

    Q. front = 23, Q = {2, 3}

    mark [3] = 1 →) vòng lặp lẻ.

Lần lặp 24

Q = {2, 3}

while (Q. front = 23 ≤ Q. rear = 24)}

    u = Q. data [23] = 2

    Q. front = 24, Q = {3}

    mark [2] = 1 →) vòng lặp lẻ

Lần lặp 25

Q = { 3}

while (Q. front = 24 ≤ Q. rear = 24)}

    u = Q. data [24] = 3

    Q. front = 25, Q = { }

    mark [3] = 1 →) vòng lặp lẻ

Lần lặp 26

Q = { }

while (Q. front = 25 > Q. rear = 24)}

    kết thúc vòng lặp

}

④