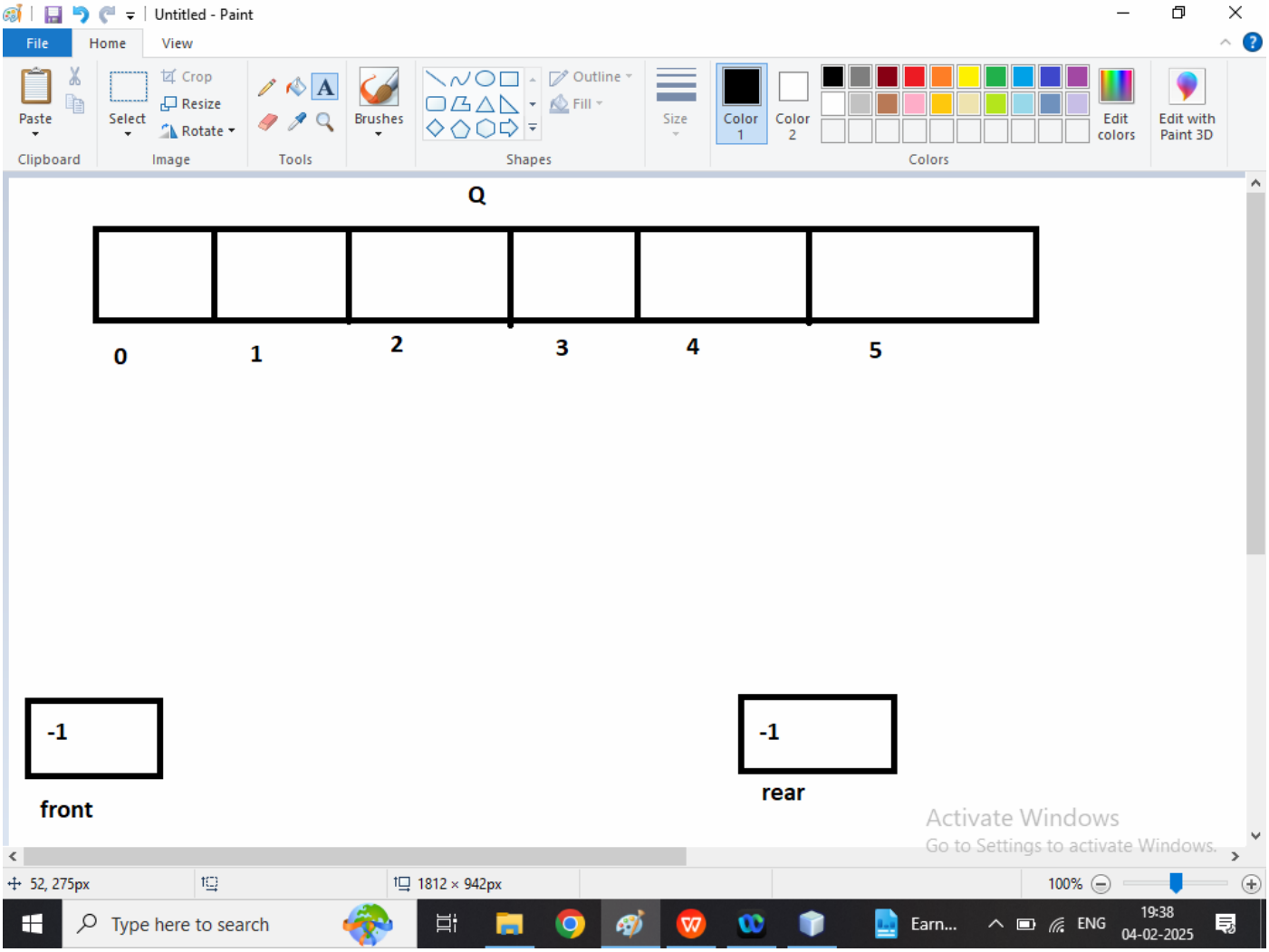
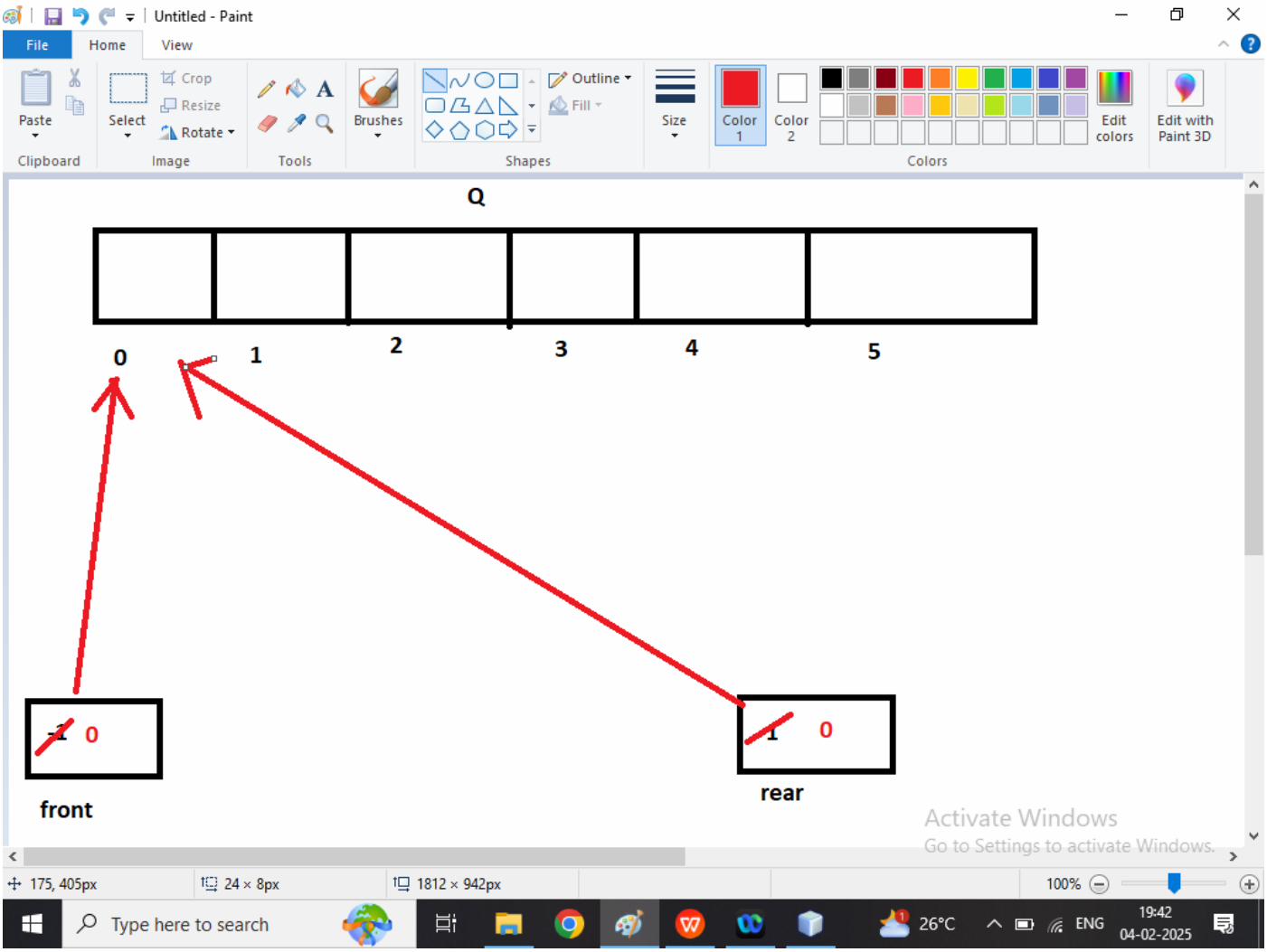
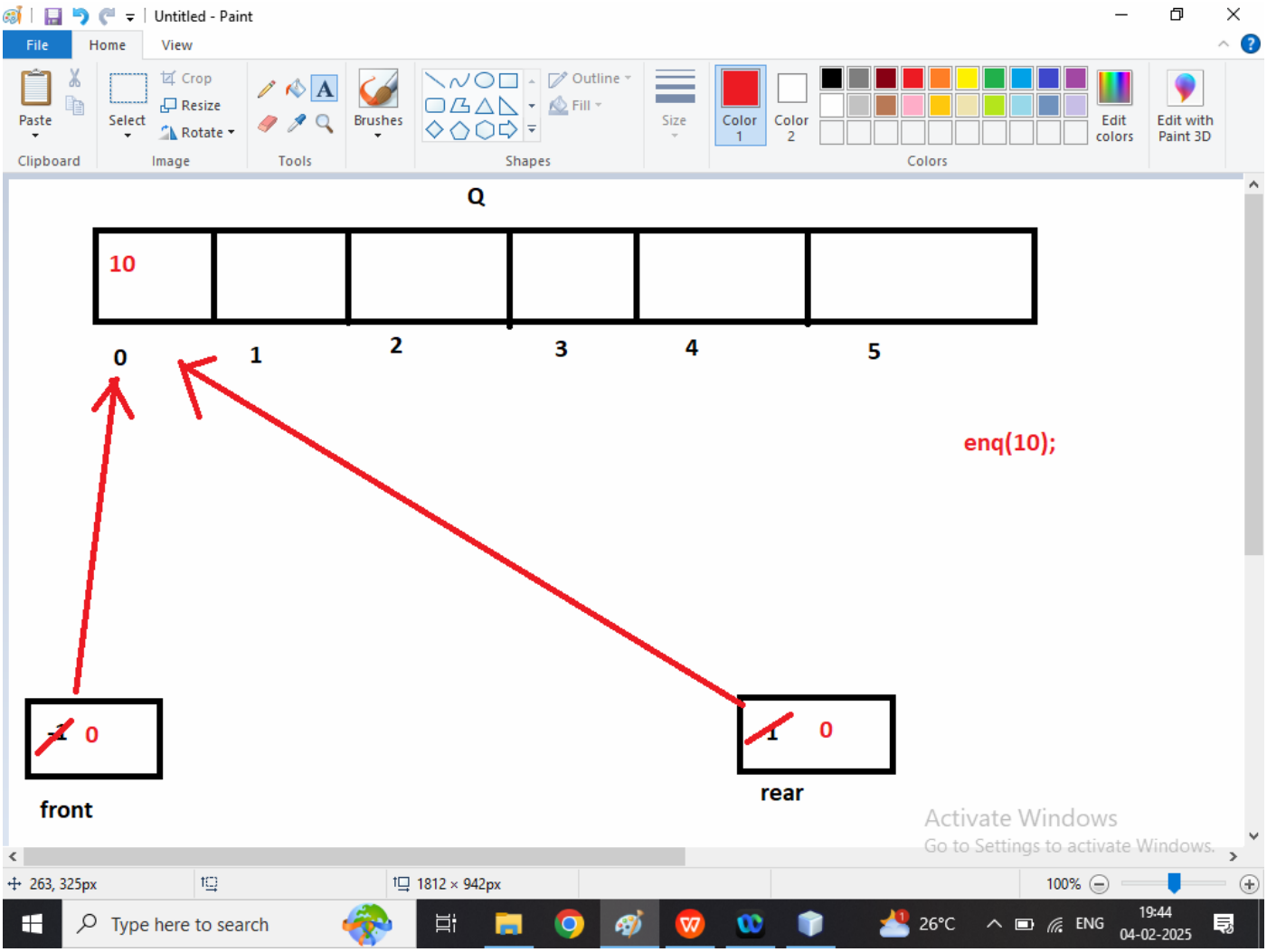
Q1. Write a java program to implement Queue data structure using array?

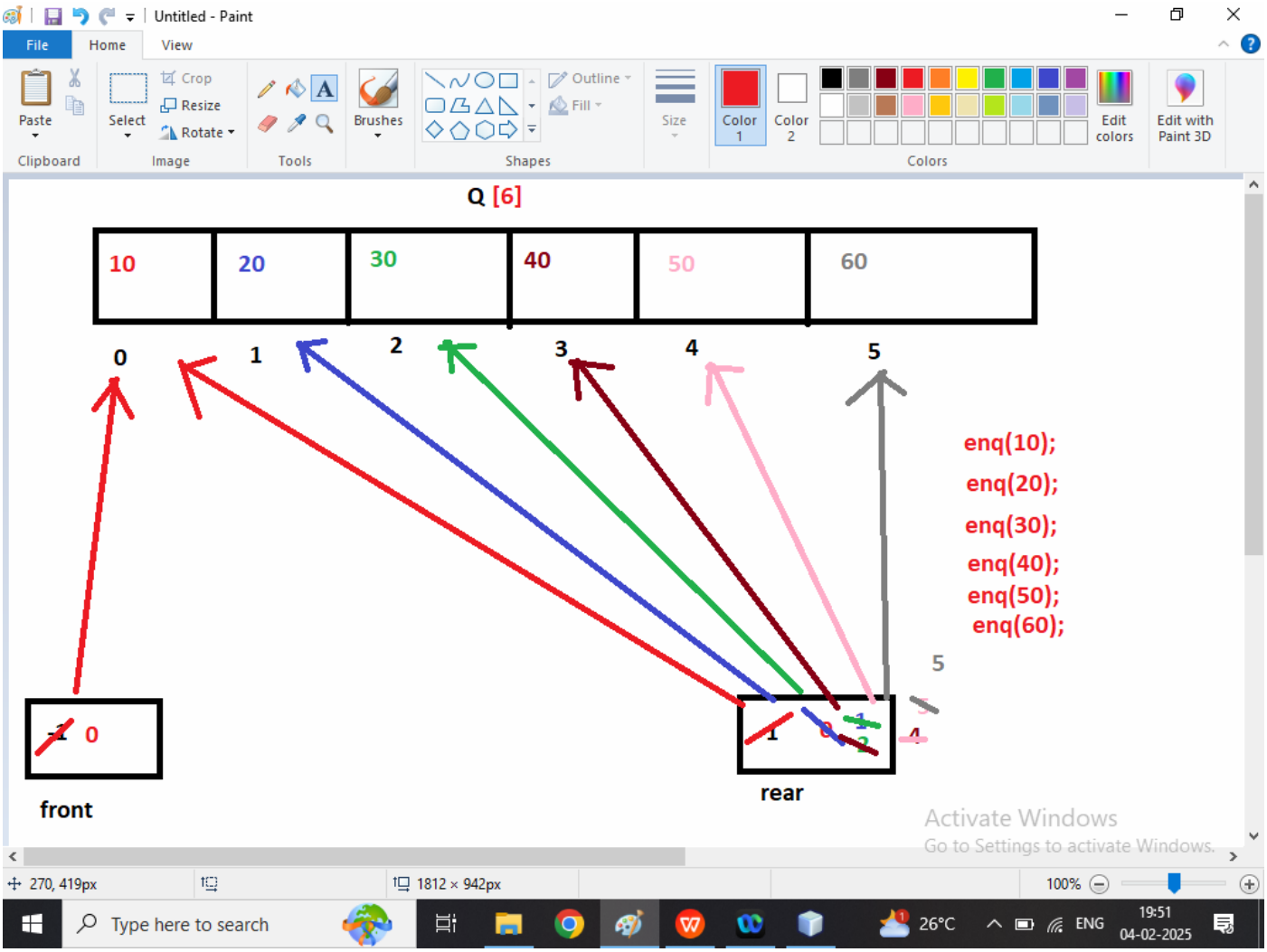


Algorithm of enque() operation of Queue Data Structure

// Step1: First Check Over flow Condition
//Step2: To check Queue is Empty Increase front and rear by 1
//Step3: If queue is Not Empty then increase rear by 1
//Step4: Insert data into queue


Untitled - Paint
File
Home
View
Paste
Select
Crop
Resize
Rotate
Brushes
Outline
Fill
Size
Color 1
Color 2
Edit colors
Edit with Paint 3D
Clipboard
Image
Tools
Shapes
Colors
Q
0
1
2
3
4
5
-1
0
front
1
0
rear
Activate Windows
Go to Settings to activate Windows.
175, 405px
24 × 8px
1812 × 942px
100%
Type here to search
26°C
ENG
19:42
04-02-2025


Untitled - Paint
File
Home
View
Paste
Select
Crop
Resize
Rotate
Brushes
Outline
Fill
Size
Color 1
Color 2
Edit colors
Edit with Paint 3D
Clipboard
Image
Tools
Shapes
Colors
Q
10
0
1
2
3
4
5
enq(10);
-1
0
front
1
0
rear
Activate Windows
Go to Settings to activate Windows.
263, 325px
1812 × 942px
100%
Type here to search
26°C
ENG
19:44
04-02-2025




```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package dsafeb2025;

public class QueueDEMO {

   static int Q[] = new int[6];
   static int front = -1;
   static int rear = -1;

   public static boolean  isEmpty(){
      return front==-1 && rear==-1;
   }
   public static boolean isFull(){
      return rear==Q.length-1;
   }
   public static void enq(int data) {
      // Step1: First Check Over flow Condition
      if(isFull()){
         System.out.println("This over flow condition");
      }else if(isEmpty()){
        //Step2: To check Queue is Empty Increase front and rear by 1
        front++;
        rear++;
```

```
            //Step4: Insert data into queue
            Q[rear]=data;
            System.out.println("First Element Insert into the queue");
        }else{
             //Step3: If queue is Not Empty then increase rear by 1
             rear++;
             Q[rear]=data;
             System.out.println("Data Insert after First Element");
        }

    }
    public static void main(String[] args) {
        enq(10);
        enq(20);
        enq(30);
        enq(40);
        enq(50);
        enq(60);
        enq(70);

    }
}
```

Q2. Write a Java Program to implemen deque operation from the Queue?

Ans:

Dequeue is the Operation used to remove an element from the front of queue. It follows first in first out order

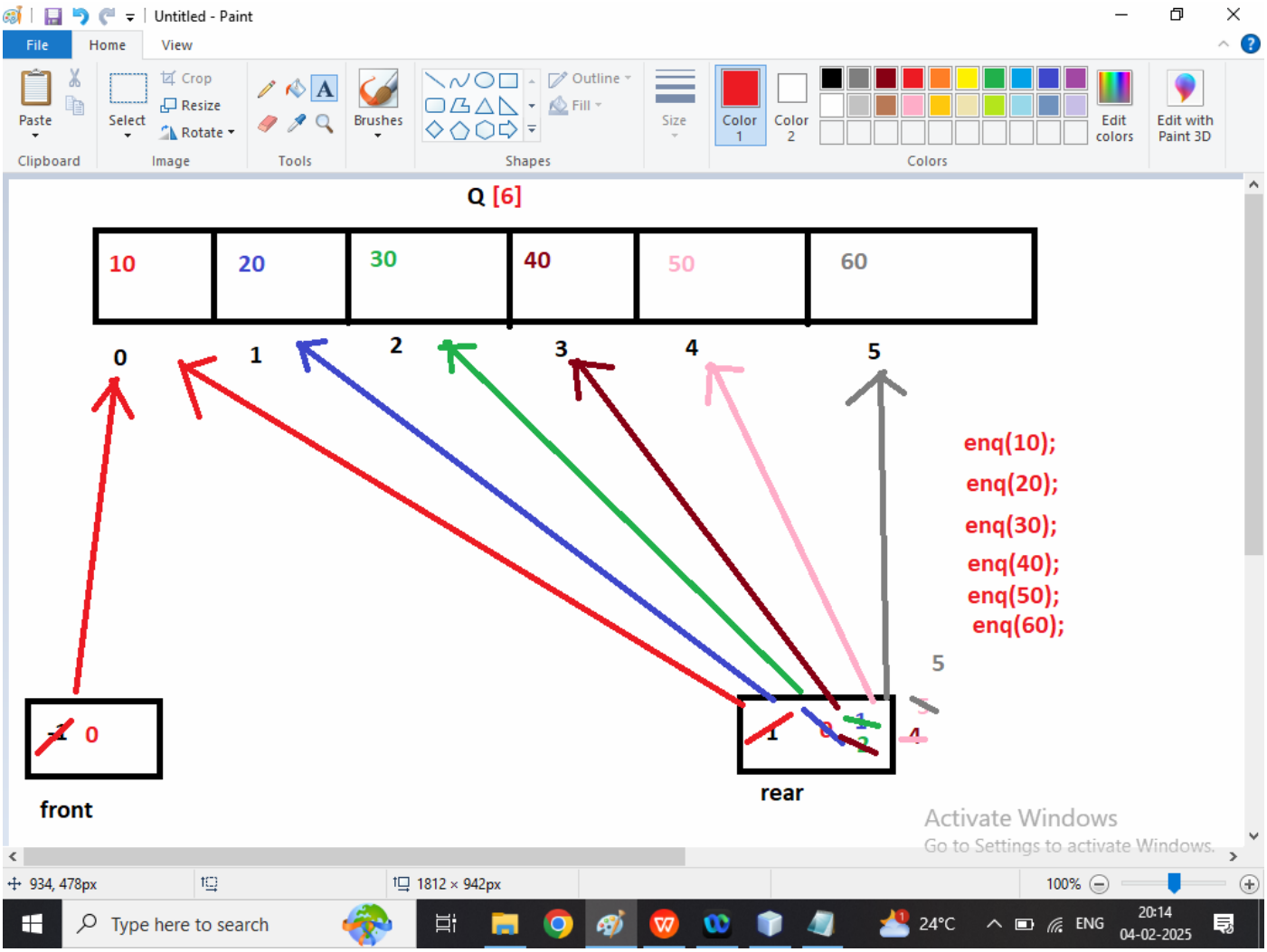
Algorithm of dequeue operation

Step1: Check Under flow condition

Step2: To check queue contains only one element so value of front assign into another variable then front and rear is -1

Step3:If queue contains more than one element store front value into another variable after that front increase by 1

Step4: Return deleted value

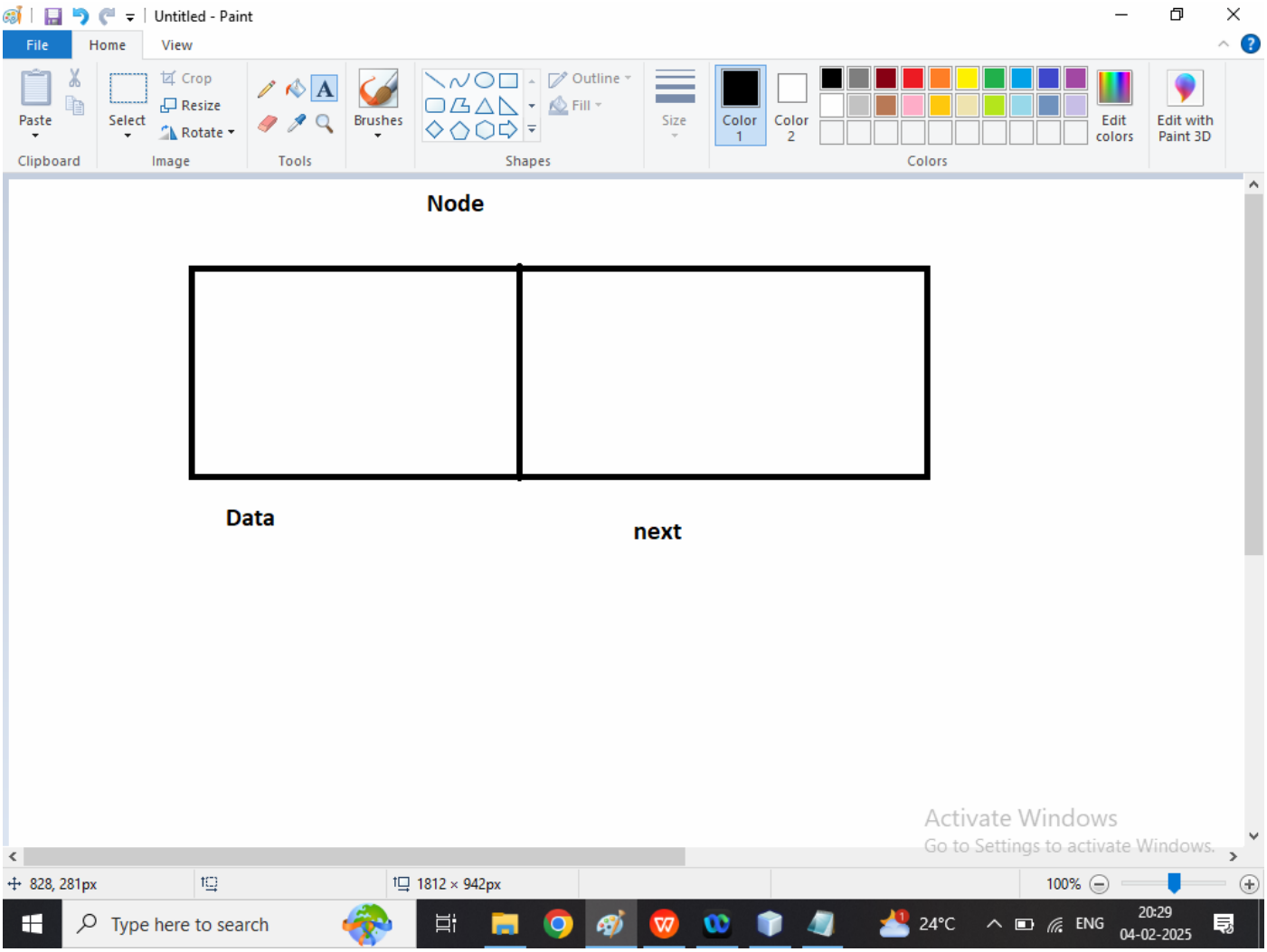



```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package dsafeb2025;

public class QueueDEMO {

   static int Q[] = new int[6];
   static int front = -1;
   static int rear = -1;

   public static boolean  isEmpty(){
      return front==-1 && rear==-1;
   }
   public static boolean isFull(){
      return rear==Q.length-1;
   }
   public static void enq(int data) {
      // Step1: First Check Over flow Condition
      if(isFull()){
         System.out.println("This over flow condition");
      }else if(isEmpty()){
        //Step2: To check Queue is Empty Increase front and rear by 1
        front++;
        rear++;


          //Step4: Insert data into queue
```

```
            Q[rear]=data;
            System.out.println("First Element Insert into the queue");
        }else{
             //Step3: If queue is Not Empty then increase rear by 1
             rear++;
             Q[rear]=data;
             System.out.println("Data Insert after First Element");
        }

    }
    public static int deq(){
        int r=0;
        //step1: To check Under flow condition
        if(isEmpty()){
            System.out.println("Under Flow Condition");
        }
        else if(rear==front){
            //step2: To check queue contains only one element so value of front assign into another variable then front and rear is -1
            r=Q[front];
            front=-1;
            rear=-1;
            System.out.println("delete last element");
        }else{
            //step3: if queue contain more than one element
            r=Q[front];
            front++;
            System.out.println("Delete element from the queue ");
        }
        //step4: Return result
        return r;
    }
    public static int  peek(){
        int r=-1;
        if(isEmpty()){
            System.out.println("Queue is Empty");
        }
        else{
            r=Q[front];
        }
        return r;
    }
    public static void main(String[] args) {
        enq(10);
        enq(20);
        enq(30);
        enq(40);
        enq(50);
        enq(60);
        enq(70);

        System.out.println("Deleted Element from the Queue  : "+deq());
        System.out.println("Deleted Element from the Queue  : "+deq());
        System.out.println("Front ========> Element "+peek());
        System.out.println("Deleted Element from the Queue  : "+deq());
        System.out.println("Deleted Element from the Queue  : "+deq());
        System.out.println("Deleted Element from the Queue  : "+deq());
        System.out.println("Deleted Element from the Queue  : "+deq());
        System.out.println("Deleted Element from the Queue  : "+deq());

    }
}
```

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package dsafeb2025;

public class QueueDEMO {

   static int Q[] = new int[6];
   static int front = -1;
   static int rear = -1;

   public static boolean  isEmpty(){
      return front==-1 && rear==-1;
   }
   public static boolean isFull(){
      return rear==Q.length-1;
   }
   public static void enq(int data) {
      // Step1: First Check Over flow Condition
      if(isFull()){
         System.out.println("This over flow condition");
      }else if(isEmpty()){
        //Step2: To check Queue is Empty Increase front and rear by 1
        front++;
        rear++;


          //Step4: Insert data into queue
          Q[rear]=data;
          System.out.println("First Element Insert into the queue");
      }else{
           //Step3: If queue is Not Empty then increase rear by 1
           rear++;
           Q[rear]=data;
           System.out.println("Data Insert after First Element");
      }

   }
   public static int deq(){
      int r=0;
      //step1: To check Under flow condition
      if(isEmpty()){
         System.out.println("Under Flow Condition");
      }
      else if(rear==front){
         //step2: To check queue contains only one element so value of front assign into another variable then front and rear is -1
         r=Q[front];
         front=-1;
         rear=-1;
         System.out.println("delete last element");
      }else{
         //step3: if queue contain more than one element
         r=Q[front];
         front++;
         System.out.println("Delete element from the queue ");
      }
      //step4: Return result
      return r;
   }
   public static int  peek(){
      int r=-1;
```

```
    if(isEmpty()){
      System.out.println("Queue is Empty");
    }
    else{
      r=Q[front];
    }
    return r;
  }
  public static void display(){
    if(isEmpty()){
      System.out.println("Queue is Empty");
    }else{
      System.out.println("Element of the Queue");
      for(int i=front;i<=rear;i++){
        System.out.print("=====>"+Q[i]);
      }
    }
  }
  public static void main(String[] args) {
    enq(10);
    enq(20);
    enq(30);
    enq(40);
    enq(50);
    enq(60);
    display();
    System.out.println("Deleted Element from the Queue  : "+deq());
    System.out.println("Deleted Element from the Queue  : "+deq());
    System.out.println("Front ========> Element "+peek());
    System.out.println("Deleted Element from the Queue  : "+deq());
    System.out.println("Deleted Element from the Queue  : "+deq());
    System.out.println("Deleted Element from the Queue  : "+deq());
    System.out.println("Deleted Element from the Queue  : "+deq());
    System.out.println("Deleted Element from the Queue  : "+deq());

  }
}
```


Untitled - Paint
File
Home
View
Paste
Select
Crop
Resize
Rotate
Brushes
Outline
Fill
Size
Color 1
Color 2
Edit colors
Edit with Paint 3D
Clipboard
Image
Tools
Shapes
Colors
Node
Data
next
Activate Windows
Go to Settings to activate Windows.
828, 281px
1812 × 942px
100%
Type here to search
24°C
ENG
20:29
04-02-2025

==========