**Designing of sorting visualizer**

1. **Bubble sort**

Create the webpage for user by html, css and bootstrap language.

**Step 1-**

Generate the size of array by taking the input from user.

Write the Javascript code for that. Create the update\_array\_size() function.

**Step 2-**

Set the speed to sorting algorithm by taking input from user.

Write the Javascript code for that. Create the vis\_speed() function.

**Step 3-**

Generate the random array by taking the input from user.

Write the Javascript code for that. Create the generate\_array() function.

**Step 4-**

Selection of the button algorithm by taking the input from user.

Write the Javascript code for that. Create the runalgo() functions to run the appropriate algorithm

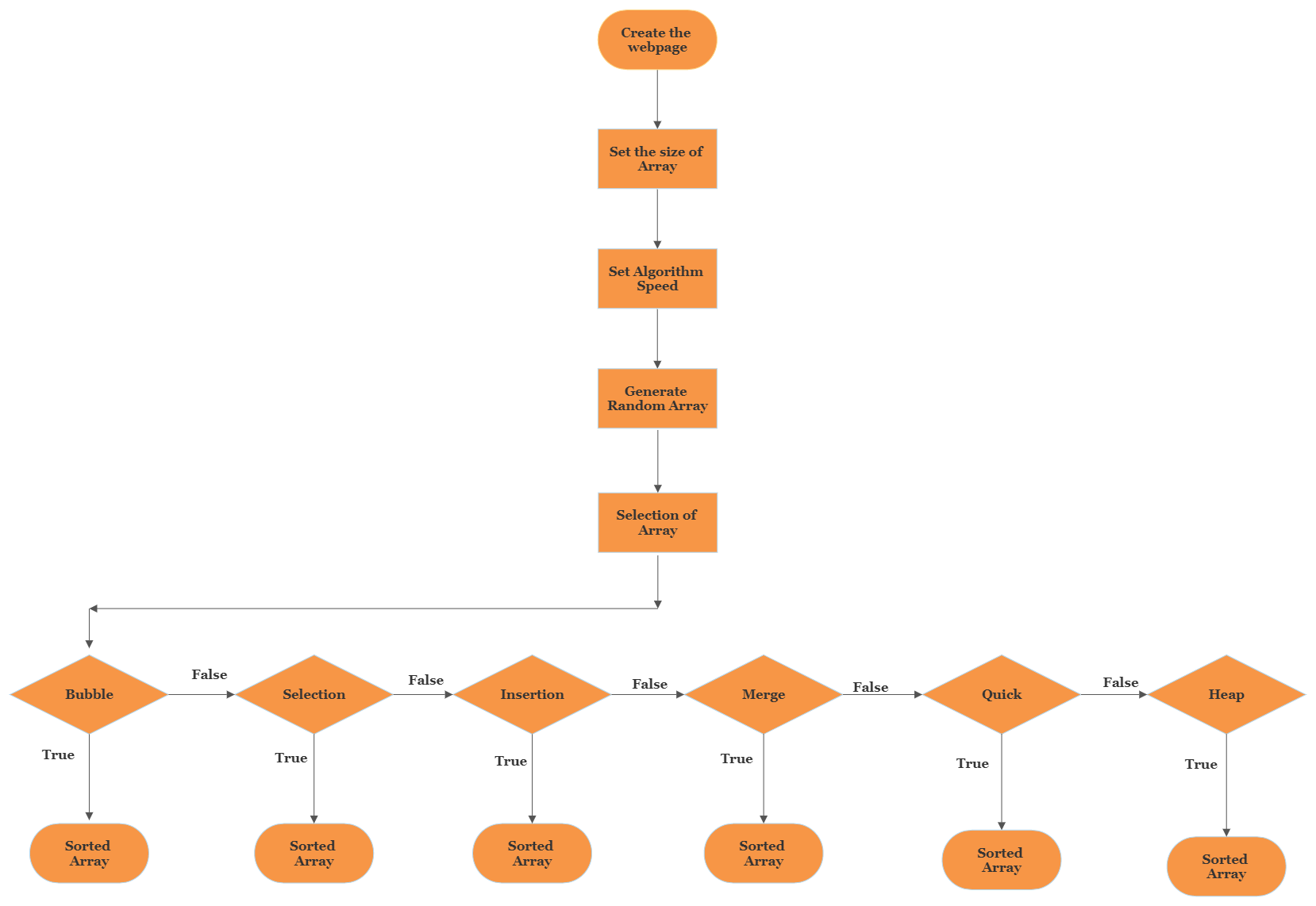
**Step 5-**

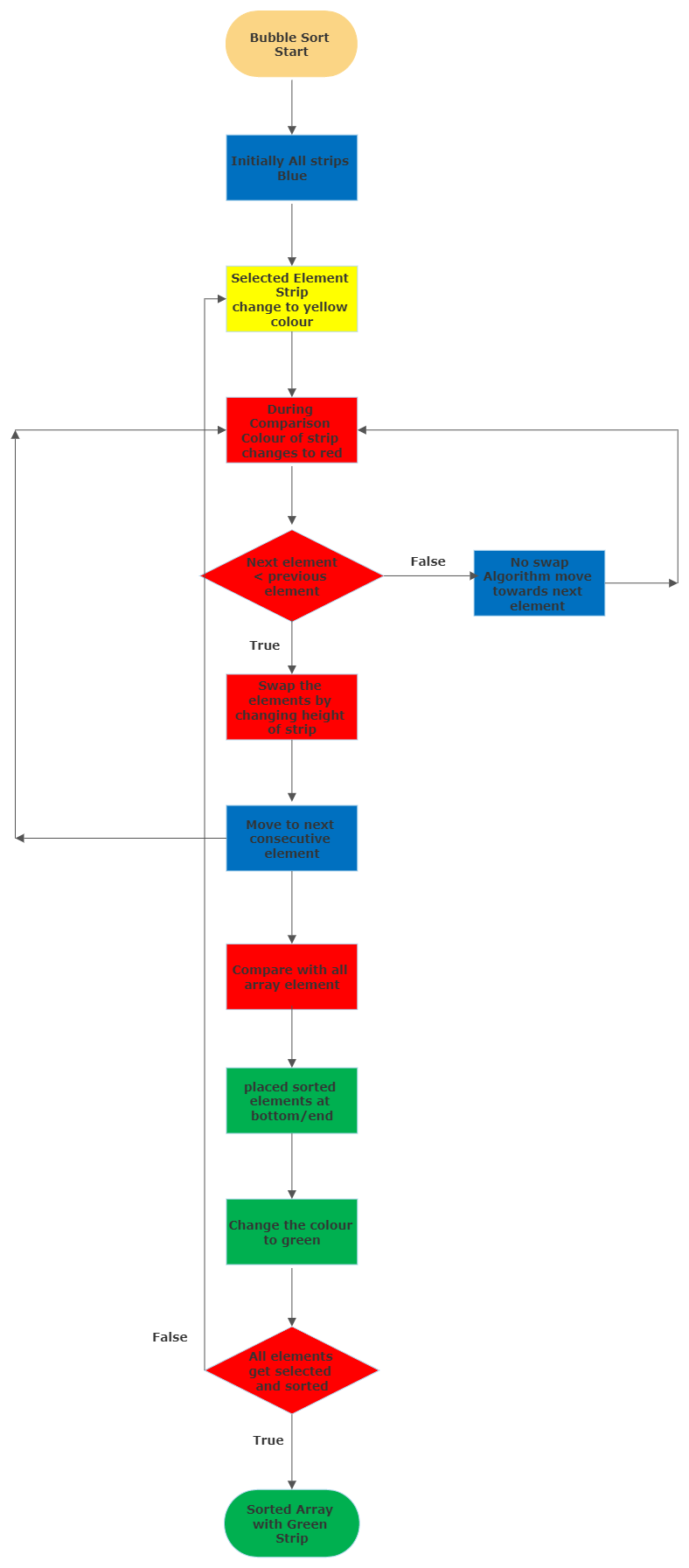
Write the code for bubble sort in javascript language.

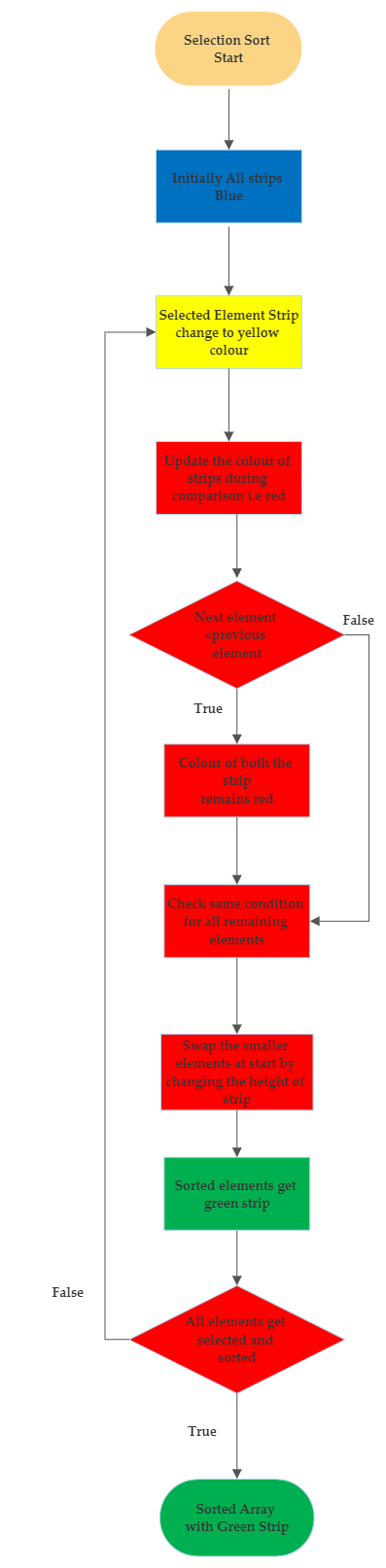
Initially all array strips are in blue color. Create the for loops for bubble sort. Select the element by changing its color to yellow. To update the color of strips create the div\_updates() function. During the comparison color of strip changes to red if first element is greater than second then swap those elements. In swaping height of the strip changes and color of strip is red. When element sorted then colour of strip is green. By bubble sort algorithm logic bigger elements placed at the end.

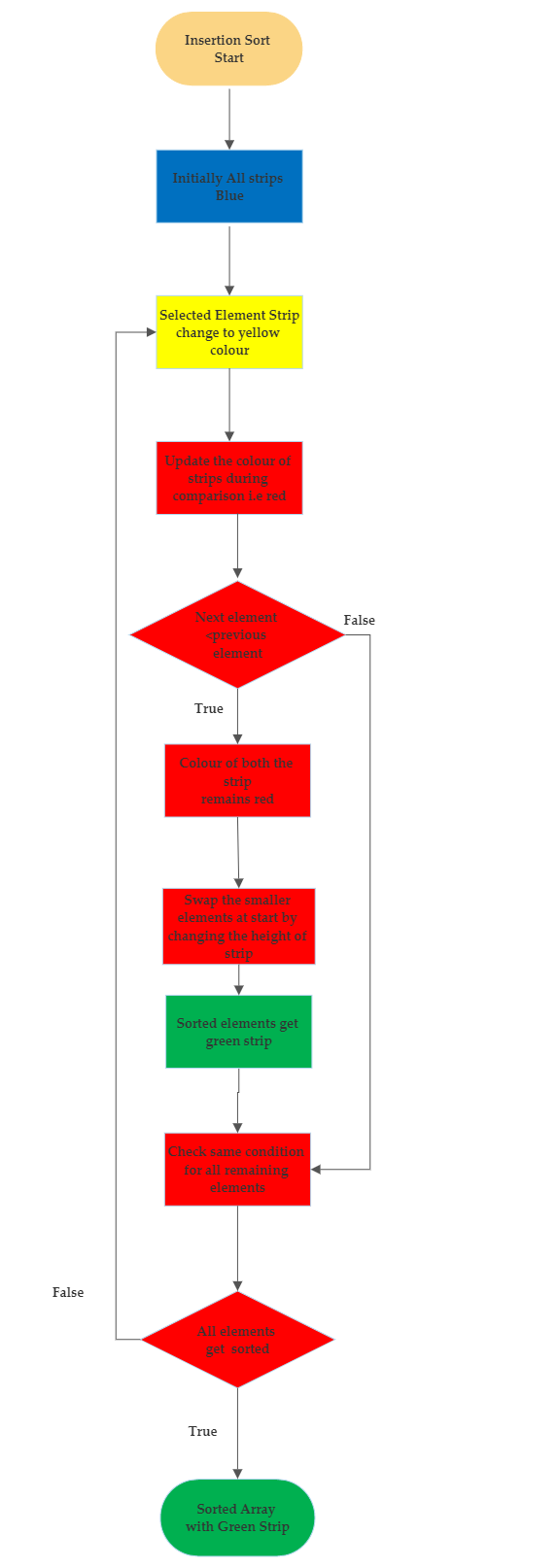
**Step 6-**

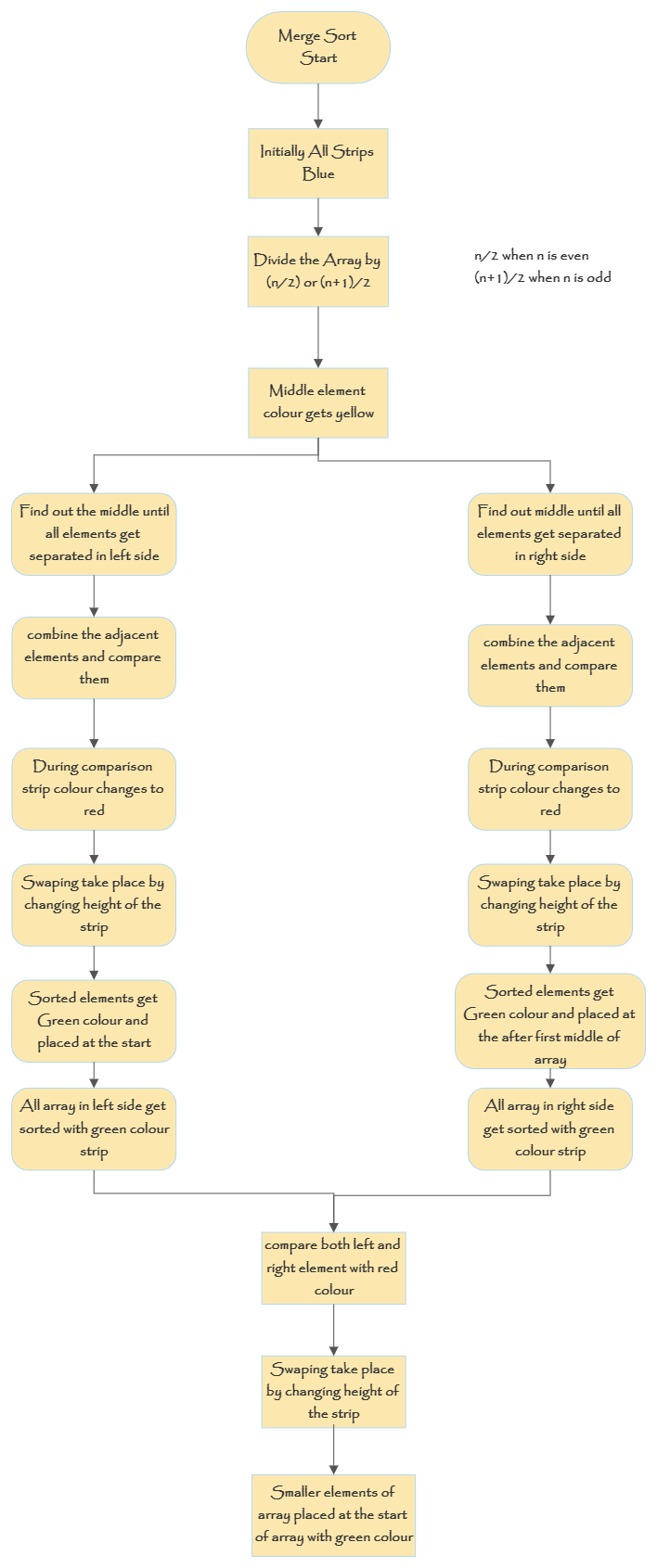
Sorted array strip in green colour strip in user website.

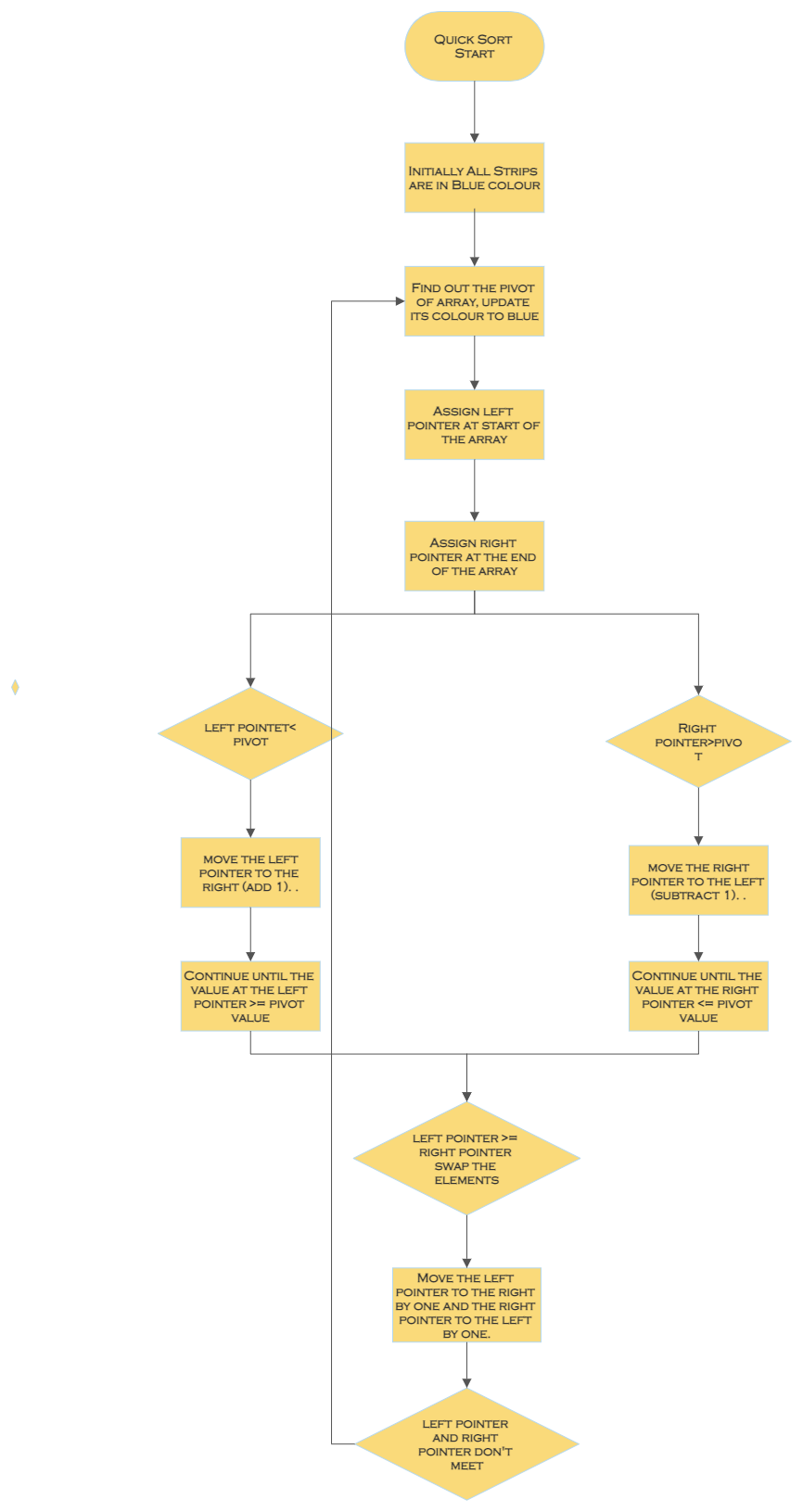
****

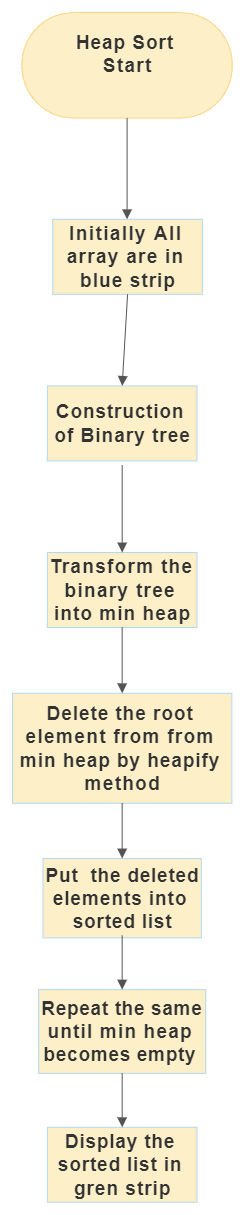
****

****

****

****

****

****

**Flow Chart**

**Create the webpage.**

**Final sorted array in green strip.**

**Swaping take place by change in height which is also in red colour. Sorted element gets green colour.**

**Element colour during selection yellow and during comparison red.**

**Bubble sort algorithm starts.**

**Generate the new array.**

**Colour of all array strips blue.**

**Set the size of array and speed of algorithm.**

1. **Selection sort**

Create the webpage for user by html, css and bootstrap language.

**Step 1-**

Generate the size of array by taking the input from user.

Write the Javascript code for that. Create the update\_array\_size() function.

**Step 2-**

Set the speed to sorting algorithm by taking input from user.

Write the Javascript code for that. Create the vis\_speed() function.

**Step 3-**

Generate the random array by taking the input from user.

Write the Javascript code for that. Create the generate\_array() function.

**Step 4-**

Selection of the selection algorithm by taking the input from user.

Write the Javascript code for that. Create the runalgo() functions to run the appropriate algorithm

**Step 5-**

Write the code for selection sort in javascript language.

In selection sort algorithm it select the first element and compare that element with all rest of the elements and replace that element. Like that algorithm takes place until all elements get sorted.

Initially all array strips are in blue color. Create the for loops for selection sort. Select the first element. To update the color of strips create the div\_updates() function.

Then select the adjacent element by updating its colour to yellow. Compare it. During the comparison colour of strip changes to red elements. If element is smaller than first selected element then colour of both strip remains same and algorithm also compare rest of elements. If it got the smaller elements then colour of the previous smaller element change to blue and new elements change to red. After comparing this with all elements. Both elements get swap by changing the heights. Then sorted element strip changes into green and another element into blue. Like that algorithm compare with all rest elements until all elements get sorted

**Step 6-**

Sorted array strip in green colour strip in user website.

**Flow Chart**

**Create the webpage.**

**Final sorted array in green strip.**

**Swaping take place by change in height which is also in red colour. Sorted element gets green colour.**

**Element colour during selection yellow and during comparison red.**

**Selection sort algorithm starts after clicking on selection button.**

**Generate the new array.**

**Colour of all array strips blue.**

**Set the size of array and speed of algorithm.**

1. **Insertion sort**

Create the webpage for user by html, css and bootstrap language.

**Step 1-**

Generate the size of array by taking the input from user.

Write the Javascript code for that. Create the update\_array\_size() function.

**Step 2-**

Set the speed to sorting algorithm by taking input from user.

Write the Javascript code for that. Create the vis\_speed() function.

**Step 3-**

Generate the random array by taking the input from user.

Write the Javascript code for that. Create the generate\_array() function.

**Step 4-**

Selection of the insertion algorithm by taking the input from user.

Write the Javascript code for that. Create the runalgo() functions to run the appropriate algorithm.

**Step 5-**

Write the code for insertion sort in javascript language.

In insertion sort algorithm comparison take place in all selected elements and elements get swap by there value. Swaping takes place in adjacent selected elements until we selected the last element of array.

Initially all array strips are in blue color. Create the for loops for selection sort. Select the first element its colour get change to yellow. To update the color of strips create the div\_updates() function.

Then select the adjacent element by updating its colour to yellow. Compare it. During the comparison colour of strip changes to red elements. In this sort comparison of selected elements takes place with each and every sorted elements and then it will get swapped like that algorithm runs until the last element

**Step 6-**

Sorted array strip in green colour strip in user website.

**Flow Chart**

**Create the webpage.**

**Final sorted array in green strip.**

**Swaping take place by change in height which is also in red colour. Sorted element gets green colour.**

**Element colour during selection yellow and during comparison red.**

**Insertion sort algorithm starts after clicking on insertion button.**

**Generate the new array.**

**Colour of all array strips blue.**

**Set the size of array and speed of algorithm.**

1. **Merge sort**

Create the webpage for user by html, css and bootstrap language.

**Step 1-**

Generate the size of array by taking the input from user.

Write the Javascript code for that. Create the update\_array\_size() function.

**Step 2-**

Set the speed to sorting algorithm by taking input from user.

Write the Javascript code for that. Create the vis\_speed() function.

**Step 3-**

Generate the random array by taking the input from user.

Write the Javascript code for that. Create the generate\_array() function.

**Step 4-**

Selection of the Merge algorithm by taking the input from user.

Write the Javascript code for that. Create the runalgo() functions to run the appropriate algorithm.

**Step 5-**

Write the code for Merge sort in javascript language.

Merge sort works on divide and conquer method. It divides the array until all elements get separated. After the separation it again start combining the elements during combination it swaps the element

Initially all array strips are in blue color. To update the color of strips create the div\_updates() function.

First divide the array by (start + end)/2 then get the middle update that into yellow color.

Again find out the new middle for left side of the middle for that take start as start and middle as end and obtain the new middle change its colour to yellow. Repeat this until all elements get separated.

After separating the all elements then start combining adjacent elements sort the elements by swaping.

Same process will be repeat for the right side of the middle element. For that take start as mid +1 and end as end.

By this will get sorted elements in left and right side of the middle

Again swap the elements by changing the heights of strip and finally will get swapped strip.

**Step 6-**

Sorted array strip in green colour strip in user website.

**Flow Chart**

**Create the webpage.**

**Final sorted array in green strip.**

**Swaping take place by change in height which is also in red colour. Sorted element gets green colour.**

**Element colour during selection yellow and during comparison red.**

**Merge sort algorithm starts after clicking on insertion button.**

**Generate the new array.**

**Colour of all array strips blue.**

**Set the size of array and speed of algorithm.**